
Conformal Online Model Aggregation

Matteo Gasparin¹

Aaditya Ramdas²

¹Department of Statistical Sciences, University of Padova, Padova, Italy

²Department of Statistics and Data Science, Carnegie Mellon University, Pittsburgh, Pennsylvania, USA

Abstract

Conformal prediction equips machine learning models with a reasonable notion of uncertainty quantification under mild assumptions. However, conformal prediction only works if we fix the underlying model in advance. A relatively unaddressed issue in conformal prediction is that of model selection and/or aggregation: given a set of prediction models, which one should we conformalize? We suggest that instead of performing model *selection*, it can be prudent and practical to perform *conformal set aggregation* in an online, adaptive fashion. We propose a wrapper that takes in several conformal prediction sets, and outputs a single adaptively-combined prediction set. Our method, called conformal online model aggregation (COMA), is based on combining the prediction sets from several algorithms by weighted voting, and can be thought of as a sort of *online stacking of the underlying conformal sets*. As long as the input sets have (distribution-free) coverage guarantees, COMA retains coverage guarantees, under a negative correlation assumption between errors and weights. COMA is well suited to decentralized settings, where users are only willing to share their prediction sets for a new test point in a black-box fashion. As we demonstrate, it is also well suited to settings with distribution drift, where model selection is not advisable.

1 INTRODUCTION

Conformal prediction has emerged as a significant approach within the field of machine learning, offering a reliable wrapper for uncertainty quantification. Unlike traditional methods that rely on strict distributional assumptions, conformal prediction provides a flexible framework by transforming

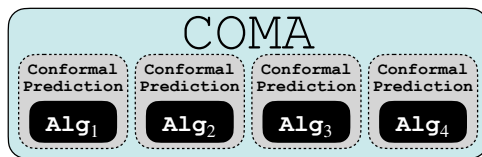


Figure 1: Graphical summary of the COMA method. Conformal prediction wraps around the individual prediction algorithms, while COMA operates as a higher-level wrapper that aggregates these sets.

point predictions obtained with any black-box model into prediction sets with valid finite-sample marginal coverage. Initially introduced by Saunders et al. [1999], conformal prediction has gained increasing prominence and has been extended beyond the iid (or exchangeable) data setting; examples include Tibshirani et al. [2019], Gibbs and Candès [2021], Barber et al. [2023].

A significant challenge in the application of conformal prediction arises in the context of model selection or aggregation. The conventional approach involves fixing the underlying prediction model in advance and then constructing prediction sets with a prespecified level of marginal coverage. Since different models yield sets with the same theoretical coverage guarantees for independent and identically distributed (iid) data, the natural goal is to select the best of a collection of models. In online settings with distribution drift, it is imprudent to commit to one model (model selection), and hence a better idea is to aggregate the models in an adaptive manner (model aggregation).

We propose a method that sequentially combines the prediction sets produced by different models through a voting mechanism, adapting the ideas of Gasparin and Ramdas [2024] to the online setting. Our method acts as a wrapper around the sets produced by conformal prediction, as illustrated in Figure 1, and can be informally viewed as a stacking of conformal sets, where the weights assigned to each model in the aggregation process are dynamically

adjusted over time to reflect their past performance. The performance of the various models is assessed in terms of set efficiency, measured by cardinality for discrete sets and Lebesgue measure for continuous sets. In particular, models that consistently produce smaller sets are favored.

We examine both iid and online scenarios with potential distributional shift, and show that our proposed solution, called COMA, performs effectively in both frameworks. In particular, in the latter case, it is used in conjunction with the adaptive conformal approaches proposed by Gibbs and Candès [2021] and Angelopoulos et al. [2024a,b]. Our solution, which is based on online learning techniques, provides a regret bound on the size of the aggregated prediction set. In addition, the final set maintains valid coverage under a negative correlation assumption between model errors and aggregation weights, an assumption that is empirically supported in our experiments. Since the proposed approach relies solely on the underlying prediction sets, it is particularly well-suited to distributed settings in which users are willing to share only their prediction sets.

1.1 RELATED WORK

Conformal prediction was introduced by Vovk and collaborators, with the first book on the topic dating back to Vovk et al. [2005]. Several other influential papers in the “offline” setting include Papadopoulos et al. [2002], Lei et al. [2018], Romano et al. [2019], Romano et al. [2020] and Barber et al. [2021]. All these approaches provide reliable uncertainty quantification for a single black-box algorithm and are grounded in the assumption that observations are iid or exchangeable. Cherubin [2019] and Solari and Djordjilović [2022] introduced a technique to merge conformal prediction sets using a simple majority voting scheme. This method was further refined by Gasparin and Ramdas [2024], who added a weighting system and the use of randomization. These procedures can be viewed as combining the corresponding conformal p-values and then inverting the resulting combined p-value to obtain the final prediction set. Although several methods have been proposed for combining p-values [Vovk et al., 2022, Toccaceli and Gammernan, 2017, Vovk, 2015, Campagner et al., 2024], or more generally for combining conformity scores from different predictors [Luo and Zhou, 2025, Alami et al., 2026, Patel et al., 2025], the majority vote procedure does not require direct access to p-values, scores or data. Instead, it only requires the prediction sets produced by the different models. In a similar spirit, Trunov and V’yugin [2023] propose a method for combining predictive distributions (rather than sets) in an online manner, although with the different goal of optimizing a probability scoring rule. Humbert et al. [2023] propose a conformal aggregation method for federated learning, where agents can share limited information due to privacy constraints.

Model selection within conformal prediction has been stud-

ied by Yang and Kuchibhotla [2025] and Fan et al. [2023]. While EFCP [Yang and Kuchibhotla, 2025] requires only prediction intervals, VFPC [Yang and Kuchibhotla, 2025] and the method of Fan et al. [2023] require access to the underlying data and models, and rely on stronger assumptions than our approach. Our work is distinct and not directly comparable: it operates in both iid and non-iid online settings, applies to classification and regression tasks, and focuses on model aggregation rather than model selection.

Recent works have extended conformal prediction beyond iid or exchangeable data [Chernozhukov et al., 2018, Tibshirani et al., 2019, Podkopaev and Ramdas, 2021, Dunn et al., 2023, Barber et al., 2023]. A related line of work considers online settings with possible continuous distributional shift. A pioneering contribution is Gibbs and Candès [2021], whose main idea is to adapt the error level α over time to achieve the desired error rate. Extensions of this method have been proposed by Zaffran et al. [2022], Gibbs and Candès [2024], Angelopoulos et al. [2024a,b].

Lastly, outside the conformal framework, the problem of combining expert advice is well studied in online learning. Typically, at each iteration, a weight is assigned to each expert and subsequently updated according to the expert’s performance. One of the most widely employed methods is the Exponential Weighted (EW) algorithm [Littlestone and Warmuth, 1994, Freund and Schapire, 1997]. These methods usually aim to minimize regret under bounded adversarial losses, and are therefore not directly suited to our setting (minimizing the size of the merged prediction sets).

2 PROBLEM SETUP

We focus our attention on an online setting in which data arrive sequentially over time. A plausible scenario arises when dealing with a continuous flow of data; in symbols, at each iteration $t = 1, 2, \dots, T$, a covariate-response pair $z^{(t)} = (x^{(t)}, y^{(t)}) \in \mathcal{X} \times \mathcal{Y}$ representing a realization of the random vector $Z^{(t)} = (X^{(t)}, Y^{(t)})$, is observed. Technically, we first observe $X^{(t)}$, then output a prediction set $\mathcal{C}^{(t)}$ that aims to contain $Y^{(t)}$, and we then observe $Y^{(t)}$.

We assume access to K different prediction algorithms (or *models* or *experts*) used to predict $Y^{(t)}$ based on covariates $X^{(t)}$. As a result, at each iteration and for a prespecified error tolerance $\alpha \in (0, 1)$, we can construct K distinct conformal prediction sets, one for each model, $\mathcal{C}_1^{(t)}, \dots, \mathcal{C}_K^{(t)}$, based on the information available in $(\{z^{(i)}\}_{i=1}^{t-1}, x^{(t)})$. In practice, due to computational constraints or potential distribution shifts, these sets may be constructed using only a subset of $\{z^{(i)}\}_{i=1}^{t-1}$. If the data are iid or exchangeable, conformal prediction sets are known to have valid marginal coverage:

$$\mathbb{P}\left(Y^{(t)} \in \mathcal{C}_k^{(t)}\right) \geq 1 - \alpha, \quad k = 1, \dots, K, \quad (1)$$

where the probability is taken with respect to all sources of randomness (the new data point, the training points, possibly the algorithm). We refer the reader to Appendix A for an introduction to conformal prediction.

The main question that arises is how to efficiently merge or select among the different conformal prediction sets. This issue is particularly relevant in distributed settings, where the aggregator has access only to the K sets, but not to the underlying data or individual model predictions. A key challenge is that these sets are not independent, since they are all constructed from the same underlying data.

We build on an approach recently proposed by Gasparin and Ramdas [2024], which involves a weighted majority vote. Let $w^{(t)} = (w_1^{(t)}, \dots, w_K^{(t)})$ be a set of weights such that

$$w_k^{(t)} \in [0, 1], \quad k = 1, \dots, K, \quad \text{and} \quad \sum_{k=1}^K w_k^{(t)} = 1,$$

where the weight $w_k^{(t)}$ represents the importance of the prediction algorithm k at time t in the voting procedure. We can define the *weighted majority vote set* as

$$\mathcal{C}_M^{(t)} := \left\{ y \in \mathcal{Y} : \sum_{k=1}^K w_k^{(t)} \mathbb{1}\{y \in \mathcal{C}_k^{(t)}\} > \frac{1 + u^{(t)}}{2} \right\}, \quad (2)$$

where $u^{(t)}$ is a realization from $U^{(t)}$ that is a random variable uniformly distributed in $[0, 1]$ independent of all the data. If the weights are chosen a priori (independent of the data), Gasparin and Ramdas [2024] prove that if the starting sets are valid (they satisfy (1)) then the coverage of $\mathcal{C}_M^{(t)}$ is at least $1 - 2\alpha$. The factor 2 is provably unavoidable and arises from the arbitrary nature of the dependence between sets in this scenario. Moreover, as pointed out in the above work, the majority vote set can be computed in $O(K \log K)$ time for real-valued sets. This is due to the need of sorting the interval endpoints and making a single pass over them. In the discrete scenario, the process is straightforward and requires only the counting of sets that include a specific label. It is worth noting that the method requires only the K input sets to construct $\mathcal{C}_M^{(t)}$. This feature is particularly advantageous in decentralized settings, where the aggregator, at each time step, has access solely to the sets themselves rather than to the conformal p-values, the scores, or the raw outputs/predictions of the individual models.

There are three hurdles to applying these ideas directly:

- First, when weights are based on past data, the weights become random variables, so existing theory does not apply. This limitation is addressed in Section 3, where we extend the theoretical results to the case where the weights themselves are random variables.
- The second question is how one should update the weights in a theoretically sound way that performs well

practically. In Section 4, we adapt an online learning method for updating the weights based on the performance of the different models. In particular, algorithms that consistently produce smaller sets are favored.

- Finally, when the data distribution shifts over time, conformal prediction no longer guarantees marginal validity. In Section 5, we extend the method and use it in conjunction with adaptive conformal prediction techniques to handle such settings.

3 WEIGHTED MAJORITY WITH DATA-DEPENDENT WEIGHTS

Suppose that the weight vector $w^{(t)}$ is a realization of a random vector $W^{(t)}$ that maps the first $t - 1$ observations of $\{Z^{(i)}\}_{i=1}^{t-1}$ into the $(K - 1)$ -dimensional simplex $\Delta_{K-1} := \{w \in [0, 1]^K : \mathbf{1}^\top w = 1\}$, or more succinctly, $W^{(t)} : (\mathcal{X} \times \mathcal{Y})^{t-1} \rightarrow \Delta_{K-1}$. In addition, we define the miscoverage indicator for the k -th set at time t as $\phi_k^{(t)} := \mathbb{1}\{y^{(t)} \notin \mathcal{C}_k^{(t)}\}$ and $[K] := \{1, \dots, K\}$. In the following, we will make the assumptions:

Assumption 1. *Marginal coverage and negative total elementwise correlation between $\phi^{(t)}$, $W^{(t)}$:*

$$\begin{aligned} \mathbb{E}[\phi_k^{(t)}] &\leq \alpha \text{ for } k \in [K], \text{ and} \\ \mathbb{E} \left[\sum_{k=1}^K \left(\phi_k^{(t)} - \mathbb{E}[\phi_k^{(t)}] \right) \left(W_k^{(t)} - \mathbb{E}[W_k^{(t)}] \right) \right] &\leq 0. \end{aligned} \quad (3)$$

While marginal coverage is guaranteed if data are iid or at least exchangeable, $\phi^{(t)}$, $W^{(t)}$ could be negatively correlated if, for example, predictors that were assigned high weights (based on previous data) are less likely to make mistakes. The assumption appears to hold in practical applications. Appendix B provides alternative conditions leading to valid coverage, including training-conditional coverage of the individual predictors. Proofs are deferred to Appendix E.

Theorem 2. *Let $\mathcal{C}_1^{(t)}, \dots, \mathcal{C}_K^{(t)}$ be $K \geq 2$ different conformal prediction sets for $Y^{(t)}$, let $W^{(t)}$ be a random vector in Δ_{K-1} depending only on $\{Z^{(i)}\}_{i=1}^{t-1}$, and consider the weighted majority vote set $\mathcal{C}_M^{(t)}$ in (2). It always holds that $\mathbb{P}(Y^{(t)} \notin \mathcal{C}_M^{(t)}) \leq 2\mathbb{E}[\max_{k \in [K]} \phi_k^{(t)}]$. If Assumption 1 holds, then we further have $\mathbb{P}(Y^{(t)} \notin \mathcal{C}_M^{(t)}) \leq 2\alpha$. This additionally implies that with probability at least $1 - 2\alpha$ the set $\mathcal{C}_M^{(t)}$ is non-empty.*

Remark 3. *The first part of the theorem states, without any assumption, that the probability of miscoverage is bounded by $2\mathbb{E}[\max_{k \in [K]} \phi_k^{(t)}]$. This can translate into a trivial bound (like 1) if there is at least one algorithm that always makes an error; while it can be approximately upper bounded by 2α in the case where miscoverage events are strongly correlated and $\mathbb{E}[\phi_k^{(t)}] \leq \alpha$ for each $k \in [K]$.*

Remark 4. *The use of randomization enhances the statistical efficiency of the prediction sets while preserving the validity of the procedure. Importantly, if one opts to avoid randomization, for instance, to mitigate any potential for p -hacking, then setting $u^{(t)} = 0$ maintains the coverage guarantee, which is ensured by Markov’s inequality.*

Remark 5. *On examining the proof, it is easy to see that the coverage guarantee deteriorates smoothly with the degree of violation of Assumption 1. In particular, if we relax the assumption in (3) to $\sum_{k=1}^K \mathbb{E}[\phi_k^{(t)} W_k^{(t)}] \leq (1 + \nu) \sum_{k=1}^K \mathbb{E}[\phi_k^{(t)}] \mathbb{E}[W_k^{(t)}]$, for some $\nu \geq 0$ which measures the degree to which assumption (3) is violated, we see that the corresponding bound on the miscoverage becomes $2\alpha(1 + \nu)$, recovering the above theorem when $\nu = 0$.*

If the weight vector takes the form $w_k^{(t)} \approx 1$, and $w_j^{(t)} \approx 0$ for all $j \neq k$, then the procedure effectively reduces to selecting the k -th model, which is the desired result when there is a clearly best model. In the regression setting, a potential drawback of the procedure is that it may return a union of disjoint intervals, even when the input sets are intervals. Nevertheless, the combined set remains an interval whenever the intersection of the K intervals is non-empty [Gasparin and Ramdas, 2024], a condition that we expect to hold in most cases. Indeed, in our experiments, the probability of obtaining an empty set or a disconnected union of sets is close to zero.

4 THE COMA META-ALGORITHM

So far, we have studied the theoretical properties of the method in the case where the weights themselves are random variables; what remains is to find a sensible and practical way to update these weights in order to improve the efficiency of our weighted majority vote set. To solve this, we employ a particular variant of the Exponential Weighted (EW) algorithm. To that end, we first define the loss functions we will use within that algorithm.

4.1 LOSS FUNCTION DEFINITION

We begin by defining the loss functions relevant to the problem under consideration: for regression tasks, the loss is quantified by the interval’s size, and for classification tasks, by the count of elements within the set. We define the loss function $\ell(\cdot)$ as

$$\ell(\mathcal{C}) := \begin{cases} \text{Lebesgue measure}(\mathcal{C}), & \text{if } \mathcal{Y} \subseteq \mathbb{R}, \\ |\mathcal{C}|, & \text{if } \mathcal{Y} = \{y_1, \dots, y_D\}, \end{cases} \quad (4)$$

where $\mathcal{C}$ denotes any set, $|\mathcal{C}|$ represents the cardinality of $\mathcal{C}$, and $\{y_1, \dots, y_D\}$ describes a discrete set with D elements (or labels). More generally, one could use a nondecreasing function of the Lebesgue measure or of the cardinality.

However, the Lebesgue measure and the cardinality provide natural canonical choices.

At each iteration, we observe losses of experts $\ell^{(t)} = (\ell_1^{(t)}, \dots, \ell_K^{(t)}) \in \mathbb{R}_+^K$ and the cumulative loss for a given model after t rounds is $L_k^{(t)} := \ell_k^{(1)} + \dots + \ell_k^{(t)}$. A key property of the majority vote is that its output sets are never larger than twice the weighted average size of the input sets.

Lemma 6. *Let $\mathcal{C}_M^{(t)}$ be the set defined in (2), $\ell_M^{(t)} = \ell(\mathcal{C}_M^{(t)})$ be the loss function defined in (4) associated with $\mathcal{C}_M^{(t)}$. For any realized values $w^{(t)}$ of $W^{(t)}$ and $u^{(t)}$ of $U^{(t)}$,*

$$\ell_M^{(t)} = \ell(\mathcal{C}_M^{(t)}) \leq 2 \sum_{k=1}^K w_k^{(t)} \ell(\mathcal{C}_k^{(t)}). \quad (5)$$

Moreover,

$$\mathbb{E}[\ell_M^{(t)}] \leq \mathbb{E}\left[\sum_{k=1}^K W_k^{(t)} \ell_k^{(t)}\right]. \quad (6)$$

From (5) and (6), it is now clear that our goal is to find a good weighting system in order to narrow down the size of $\mathcal{C}_M^{(t)}$; in particular, we aim to assign a higher weight to experts that consistently yield smaller sets compared to others. Furthermore, we want the cumulative loss of our proposed method $L_M^{(t)} := \ell_M^{(1)} + \dots + \ell_M^{(t)}$ to be not much larger than (and possibly smaller than) the cumulative loss incurred by the best expert in hindsight $L_*^{(t)} := \min_{k \in [K]} L_k^{(t)}$. The improvement in the multiplicative factor in (6) relative to (5) can be interpreted as the gain due to randomization.

4.2 EMPLOYING ADAHEDGE WITHIN COMA

A way to update weights is to use the EW (or Hedge) Algorithm [Littlestone and Warmuth, 1994, Freund and Schapire, 1997], which corresponds to Algorithm 1 with $\eta^{(t)} = \eta$, for all t . The algorithm adjusts the weights based on the performance of the methods in various rounds, with adjustments made in inverse proportion to the exponential of their total loss scaled by a parameter η . This “learning rate” η plays a crucial role; if η approaches zero, the weights approach a uniform weighting, while if $\eta \rightarrow \infty$ the algorithm reduces to the Follow-the-Leader strategy, which puts all the weight on the method with the smallest loss so far.

In certain scenarios, determining the appropriate value for this parameter may be difficult; indeed, we do not know if there is a method outperforming the others or whether any constant works well in some situations but not in others. Before presenting a solution for the problem, we define two key quantities. The first is the *hedge* loss $H^{(t)} = h^{(1)} + \dots + h^{(t)}$, where $h^{(t)}$ is $h^{(t)} = w^{(t)} \cdot \ell^{(t)} = \sum_k w_k^{(t)} \ell_k^{(t)}$. This quantity can be interpreted as the expected loss achieved by

Algorithm 1 Conformal Online Model Aggregation

Require: $\mathcal{C}_1^{(t)}, \dots, \mathcal{C}_K^{(t)}$ at each t , learning rate $\eta^{(1)} \geq 0$

- 1: $w_k^{(1)} \leftarrow 1/K, L_k^{(0)} \leftarrow 0, k \in [K]$
- 2: **for** rounds $t = 1, \dots, T$ **do**
- 3: $\mathcal{C}_M^{(t)} \leftarrow \left\{ y \in \mathcal{Y} : \sum_{k=1}^K w_k^{(t)} \mathbb{1}\{y \in \mathcal{C}_k^{(t)}\} > \frac{1+u^{(t)}}{2} \right\}$
- 4: Receive loss $\ell_k^{(t)}$, update $L_k^{(t)} := L_k^{(t-1)} + \ell_k^{(t)}$
- 5: Update learning rate $\eta^{(t+1)}$
- 6: $w_k^{(t+1)} \leftarrow \exp\{-\eta^{(t+1)} L_k^{(t)}\} / \sum_{j=1}^K \exp\{-\eta^{(t+1)} L_j^{(t)}\}$
- 7: **end for**

randomly selecting the model k with probability $w_k^{(t)}$. The second is the *mix* loss

$$m^{(t)} := -\frac{1}{\eta^{(t)}} \ln \left(w^{(t)} \cdot \exp\{-\eta^{(t)} \ell^{(t)}\} \right),$$

which corresponds to the loss of an exponentially weighted aggregate predictor. Our solution is to employ the AdaHedge algorithm, introduced in de Rooij et al. [2014], which dynamically adjusts the learning parameter over time. At each round, weights are assigned according to $w_k^{(t+1)} \propto e^{-\eta^{(t+1)} L_k^{(t)}}$, $k \in [K]$, with

$$\eta^{(t+1)} := \frac{\ln K}{\delta^{(1)} + \dots + \delta^{(t)}},$$

where $\delta^{(t)} = h^{(t)} - m^{(t)}$.

The AdaHedge algorithm provides an upper bound on regret, defined as the difference between the hedge loss and the loss of the best method up to time t , specifically: $H^{(t)} - L_*^{(t)}$. Consider $\ell_+^{(t)} = \max_{k \in [K]} \ell_k^{(t)}$ and $\ell_-^{(t)} = \min_{k \in [K]} \ell_k^{(t)}$ to represent the largest and smallest loss in round t , and let $L_+^{(t)}$ and $L_-^{(t)}$ represent their cumulative sum. Additionally, let $V^{(t)} := \max\{v^{(1)}, \dots, v^{(t)}\}$ denote the largest loss range, where $v^{(t)} = \ell_+^{(t)} - \ell_-^{(t)}$. We now bound the cumulative loss of the sets produced by the weighted majority vote method.

Theorem 7. *If Assumption 1 holds, then the COMA algorithm satisfies $\mathbb{P}(Y^{(t)} \in \mathcal{C}_M^{(t)}) \geq 1 - 2\alpha$. The loss of the weighted majority vote set over T rounds is bounded as*

$$L_M^{(T)} \leq 2L_*^{(T)} + 4 \left(V^{(T)} \ln K \frac{(L_+^{(T)} - L_*^{(T)})(L_*^{(T)} - L_-^{(T)})}{L_+^{(T)} - L_-^{(T)}} \right)^{1/2} + 2V^{(T)} \left(\frac{16}{3} \ln K + 2 \right). \quad (7)$$

Further, if the loss range is bounded, i.e. $\sup_t v^{(t)} \leq D$,

$$\frac{L_M^{(T)}}{T} \leq \frac{2L_*^{(T)}}{T} + o(1). \quad (8)$$

The result in (7) is identical to the regret bound introduced in de Rooij et al. [2014], except for a multiplicative factor of 2.

In addition, the bounded loss range assumption is naturally satisfied in the classification setting. Although AdaHedge was originally developed for aggregating point forecasts, we adapt it here to the aggregation of prediction sets. Interestingly, a regret bound of the same type can still be derived in this different setting. Clearly, other methods are available to update the weights. However, we recommend the proposed solution because it does not require tuning the aggregation learning rate (it is completely data-driven), ensures a regret bound even in the case of unbounded losses, and demonstrates strong empirical performance. When our merging algorithm is used (with constant, adaptive or another step-size rule) in this conformal context, we call the method “conformal online model aggregation”, or COMA for short.

Remark 8. *The bound in Theorem 7 applies both to the deterministic and randomized versions of COMA (Remark 4). For the randomized procedure, Lemma 6 yields the sharper inequality $\mathbb{E}[\ell_M^{(t)} | W^{(t)}, \{\mathcal{C}_k^{(t)}\}_{k=1}^K] \leq h^{(t)}$, where the expectation is taken only with respect to the auxiliary randomization $U^{(t)}$. Consequently, replacing $\ell_M^{(t)} \leq 2h^{(t)}$ by this inequality in the proof of Theorem 7 shows that the corresponding regret bound holds conditionally in expectation over $U^{(t)}$, but without the multiplicative factor of 2 in (7).*

In certain instances, the Lebesgue measure of the set cannot be directly employed because of situations where the interval corresponds to the entire real line. For example, we will see that in adaptive conformal prediction [Gibbs and Candès, 2021], it may happen that the predictive set of an agent coincides with $\mathcal{Y}$. A possible solution is to define the loss function as an increasing, nonnegative, bounded function of the measure of the set (so that the weight of a predictor does not get permanently set to zero if its loss happens to be unbounded in some round). See, for example, Section 5.2.

5 COMA UNDER DISTRIBUTION SHIFT

Up to this point, we have focused on the iid setting and described the properties of COMA in this scenario. As we will see in Section 6, which reports the experimental results, in the iid setting COMA tends to concentrate its weights on the best-performing model. However, what occurs if distributional shifts arise and different models exhibit varying levels of performance at different points in time? First of all, as discussed in Section 2, conformal prediction provides a general framework to quantify uncertainty; however, its validity is based on the iid assumption (or exchangeability). This assumption is violated when the underlying data distribution shifts over time and standard conformal inference can no longer be expected to yield reliable guarantees. To address this challenge, several methods have been proposed in recent years, including Gibbs and Candès [2021] and Angelopoulos et al. [2024a]. We now briefly outline their approaches before extending our COMA procedure to this

setting. We will see that COMA can be used in combination with these approaches, serving as a flexible tool in the considered scenario. For clarity of exposition, we first introduce the methods in the case of a single algorithm, deferring the multi-expert extensions to Section 5.1 and Section 5.2.

In Gibbs and Candès [2021], *adaptive conformal inference* (ACI) is introduced as a novel method for constructing prediction sets in an online manner, providing robustness to changes in the marginal distribution of the data. Their work is based on the update of the error level α to achieve the desired level of confidence. Indeed, given the possible non-stationary nature of the data-generating distribution, conventional results do not guarantee $1 - \alpha$ coverage. However, at each time t , an alternative value $\alpha^{(t)}$ might exist, allowing the attainment of the desired coverage.

The procedure described in Gibbs and Candès [2021] recursively updates the error rate α as follows:

$$\alpha^{(1)} = \alpha, \quad (9)$$

$$\alpha^{(t)} = \alpha^{(t-1)} + \gamma(\alpha - \phi^{(t-1)}), \quad t \geq 2, \quad (10)$$

where $\phi^{(t)} = \mathbb{1}\{y^{(t)} \notin \mathcal{C}^{(t)}(\alpha^{(t)})\}$ is the sequence of mis-coverage events setting the miscoverage rate to $\alpha^{(t)}$, while $\gamma > 0$ represents a step size parameter. We refer to Appendix A for a more detailed explanation of the method. A drawback arises from the fact that in certain situations, ACI can return either $\mathcal{Y}$ or $\emptyset$ as a set, attributed to the circumstance that in some iterations, the parameter $\alpha^{(t)}$ can be smaller than 0 or greater than 1.

Another method is presented in Angelopoulos et al. [2024a], where instead of updating the error level α , there is an update of the quantile of the distribution of the *scores*. In this setting, we define $s^{(t)} : \mathcal{X} \times \mathcal{Y} \rightarrow \mathbb{R}$ as the *conformal score*, which measures the accuracy of the prediction at time t . If different models are involved one can use different scores (e.g. residual or density based scores). The procedure, starting from an initial threshold $q^{(1)} = q$, updates the quantiles as follows:

$$q^{(t)} = q^{(t-1)} + \gamma(\phi^{(t-1)} - \alpha), \quad t \geq 2,$$

where, in this case, $\phi^{(t)} = \mathbb{1}\{y^{(t)} \notin \mathcal{C}^{(t)}(q^{(t)})\}$ and the sets at each time are simply defined by

$$\mathcal{C}^{(t)}(q^{(t)}) = \left\{ y \in \mathcal{Y} : s^{(t)}(x^{(t)}, y) \leq q^{(t)} \right\}. \quad (11)$$

As noted in Angelopoulos et al. [2024a], both ACI and quantile tracking can be rephrased as an online descent algorithm with respect to the pinball loss [Koenker, 2005]. In addition, both methods achieve a *long-run coverage* in time, defined as

$$\frac{1}{T} \sum_{t=1}^T \phi^{(t)} = \alpha + o(1), \quad (12)$$

under weak or no assumptions on $\{z^{(t)}\}_{t=1}^T$ and $\{s^{(t)}\}_{t=1}^T$, where $o(1)$ is a quantity that tends to zero as $T \rightarrow \infty$. Equation (12) provides a deterministic asymptotic guarantee rather than a probabilistic marginal one. This makes long-run average coverage a natural target in online settings where the guarantee in (1) may be unattainable, although it does not imply validity at each individual time step.

As the data distribution shifts, the model weights may also vary across iterations, since different models may be preferable at different time points. This feature makes our method particularly attractive. We describe two different ways to combine our dynamic ensembling algorithm with ACI: either as a wrapper that does not alter the inner workings of ACI (Section 5.1), or by altering the feedback provided to ACI itself (Section 5.2). The corresponding algorithms are given in Appendix C.

5.1 DECENTRALIZED COMA UNDER DISTRIBUTION SHIFT

In the scenario described above, suppose that each agent operates independently by providing an interval at each iteration and, subsequently, the *aggregator* merges these intervals according to the observed vector $w^{(t)}$. This framework can be referred to as a decentralized dynamic merging. In this setting, the advantage of the quantile tracking method is that it never returns sets of infinite measure. This implies that the loss function used can remain the size of the different sets, ensuring that the bound described in (7) is valid. In the case of using the described ACI approach, it becomes necessary to employ a function $g(\cdot)$ that is capable of returning a finite number when $\alpha_k^{(t)} \leq 0$. Before proceeding with the rest of the section, we introduce a proposition which holds under the following assumptions:

1. Negative total elementwise empirical correlation between $\{\phi^{(t)}\}_{t=1}^T$ and $\{w^{(t)}\}_{t=1}^T$:

$$\frac{1}{T} \sum_{t=1}^T \sum_{k=1}^K (\phi_k^{(t)} - \bar{\phi}_k)(w_k^{(t)} - \bar{w}_k) \leq 0, \quad (13)$$

where $\bar{\phi}_k := \frac{1}{T} \sum_{t=1}^T \phi_k^{(t)}$ and $\bar{w}_k := \frac{1}{T} \sum_{t=1}^T w_k^{(t)}$.

2. No randomization: the value $u^{(t)}$ in (2) equals 0:

$$u^{(t)} = 0, \quad t = 1, \dots, T. \quad (14)$$

Proposition 9. *Let $\{z^{(t)} = (x^{(t)}, y^{(t)})\}_{t=1}^T$ be an arbitrary sequence of data points, $\{w^{(t)} = (w_1^{(t)}, \dots, w_K^{(t)})\}_{t=1}^T$ be a sequence of weight vectors, and $\{\phi_k^{(t)}\}_{t=1}^T$ be the sequence of miscoverage events for the k -th algorithm, $k \in [K]$. Under assumptions (13) and (14), for all $\alpha \in (0, 1/2]$, decentralized COMA in conjunction with adaptive conformal inference (10) satisfies*

$$\frac{1}{T} \sum_{t=1}^T \mathbb{1}\{y^{(t)} \notin \mathcal{C}_M^{(t)}\} \leq 2\alpha + 2 \frac{1 - \alpha + \gamma}{T\gamma}. \quad (15)$$

Similar results can be obtained using the quantile tracking method under the assumption of bounded scores (see Proposition 1 in Angelopoulos et al. [2024a] and Theorem 1 in Angelopoulos et al. [2024b]). Randomization is not included because the above is a deterministic result; however, the empirical results using randomization appear to be reasonably good also in this setting. The result (15) guarantees long-run coverage at least of level $1 - 2\alpha$ for any distribution, provided specific conditions about the observed weights and miscoverage errors are met. In our case, the weights will be obtained using the method described in Section 4. Although the assumption in (13) is difficult to establish rigorously, we will see that it is satisfied in practice.

5.2 ACI WITH DYNAMIC MERGING

The approach described in the preceding section had agents updating the quantiles of their score function based on *their own* past errors. This makes sense when all agents can observe the ground truth and calculate their own errors, and they do not care about the aggregator’s goals. In other words, the dynamic merging algorithm was just an outer wrapper that did not interfere with the inner functioning of the K adaptive conformal inference algorithms.

We show that our dynamic ensembling method can provide direct feedback to the adaptive conformal inference framework, and this can have some advantages in settings where the end goal is good aggregator performance only, and individual agents do not have their own goals (of maintaining coverage, say). First, we observe that in regression settings if the absolute residuals are employed as the score function and if the quantile ($q^{(t)}$) is common across all algorithms, then the size of the sets at time t is equal for all intervals. This implies that the weights are not updated during the various iterations. In this case, we propose using a value $\alpha^{(t)}$ that is common for all K agents, and it is updated by the aggregator based on the performance of the set obtained by the exponential weighted majority vote. In other words, the aggregator, at each iteration, tries to learn the coverage level required to obtain a $(1 - \alpha)$ -prediction set. The procedure updates the α -level according to the miscoverage event $\phi^{(t)} = \mathbb{1}\{y^{(t)} \notin \mathcal{C}_M^{(t)}(\alpha^{(t)})\}$, where

$$\mathcal{C}_M^{(t)}(\alpha^{(t)}) := \left\{ y \in \mathcal{Y} : \sum_{k=1}^K w_k^{(t)} \mathbb{1}\left\{ y \in \mathcal{C}_k^{(t)}(\alpha^{(t)}) \right\} > \frac{1}{2} \right\},$$

and $w_k^{(t)}$ are the weights learned by the COMA procedure. The error level (common among agents) is updated as in (9) and (10). It is important to note that the interval lengths can be different between the various agents even if $\alpha^{(t)}$ is common. In such a scenario, ACI operates directly on the set produced by the aggregator, and so we directly recover the properties of the ACI method. This is due to the fact that the sets are nested with respect to the parameter $\alpha^{(t)}$ and the

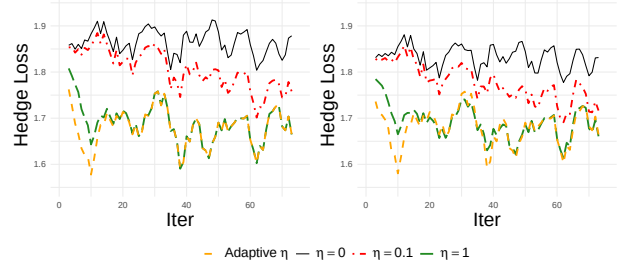


Figure 2: Hedge loss (h_t) under fixed and adaptive learning rates. The case $\eta=0$ corresponds to the standard unweighted majority vote. The left and right panels show the cases $K=4$ and $K=5$, respectively. The sequences are smoothed with a moving average $(5, \frac{1}{5})$. In both cases, COMA with adaptive η quickly achieves the smallest loss.

set is $\mathcal{Y}$ if $\alpha^{(t)} \leq 0$ and $\emptyset$ if $\alpha^{(t)} \geq 1$ (see Szabadvary and Lofstrom [2025]). Moreover, randomization is not needed to improve statistical efficiency. This scenario is plausible when agents can collaborate and the aggregator can update both the weights and the miscoverage rate.

6 EXPERIMENTAL RESULTS

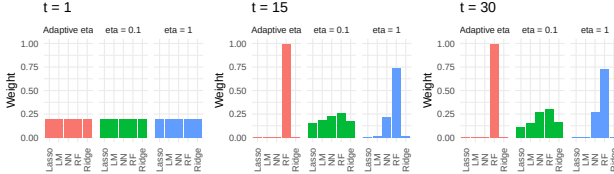
We evaluate the proposed approaches across several scenarios. In the iid setting, it is often the case that our adaptive stepsize update rule results in the weight vector rapidly putting almost full weight on the best predictor, resulting effectively in model “selection”. In contrast, under distribution shifts, the weights can meaningfully fluctuate across experts, favoring those that perform best at different times. Additional experiments are reported in Appendix D.

6.1 EXPERIMENTS IN IID SETTING

Real Data. To examine the practical behavior of COMA, we apply it to a real-world dataset. Specifically, the dataset comprises 75 075 observations pertaining to Airbnb apartments in New York City [Bernardi and Guidolin, 2023]. The response is the logarithm of the nightly price, with covariates describing the apartment and its location. The data are split into 75 rounds, each comprising 500 training observations, 500 calibration observations, and one test point. At each round, K split conformal sets are constructed using different base models. In the first simulation, $K=4$ algorithms are employed: linear model, lasso regression with penalty parameter set to 0.1, ridge regression with penalty parameter set to 0.1, and random forest with 250 trees. In the second scenario, a neural net with 8 nodes is added. In both cases, the random forest emerges as the best model, producing narrower sets. However, whereas the differences with other models are substantial in the first scenario, in the

Table 1: Size ($\pm$ sd) and coverage on real data ($\alpha=0.05$).

Method	Size	Coverage
RF (Best Single)	1.676 ± 0.010	0.973
Ensemble	1.738 ± 0.011	0.973
Bonferroni	1.863 ± 0.025	0.987
Union	2.239 ± 0.022	0.987
EFCP	1.639 ± 0.010	0.973
VFCP	1.743 ± 0.014	0.973
Ada. COMA ($K=4$)	1.669 ± 0.011	0.973
Ada. COMA ($K=5$)	1.640 ± 0.016	0.973

Figure 3: Weights assigned to the regression algorithms across iterations for the case $K=5$. The adaptive- η strategy quickly concentrates the mass on random forest.

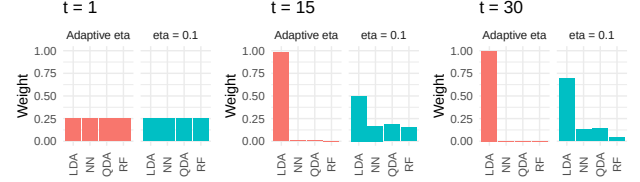
second case the neural net also delivers satisfactory results.

The weights are adjusted using the COMA method with either adaptive or fixed values of η . For Algorithm 1, when η is kept constant across iterations, two values are considered: $\eta = 0.1$ and $\eta = 1$. The case $\eta = 0$ is included as a benchmark; it corresponds to the solution proposed by Cherubin [2019], where the weights remain uniform and fixed over time. In both cases, the loss of COMA with adaptive η decreases rapidly and then stabilizes (Figure 2), as the algorithm quickly assigns unit mass to the best-performing model, namely the random forest (Figure 3). Thus, when one model clearly dominates, adaptive COMA effectively performs model selection. In contrast, with fixed $\eta = 0.1$, the loss decreases more slowly, reflecting more conservative weight updates; when the weights remain constant ($\eta = 0$), the loss is stable and no improvement occurs.

We compare our approach with several alternative methods. As a first benchmark, we apply conformal prediction to an ensemble model constructed from the $K=5$ algorithms used in our analysis, where predictions are aggregated using a simple average. Although this approach guarantees valid coverage, it requires access to the estimated outputs of the models. A second competitor is the union of the K sets. This strategy also guarantees valid coverage, but typically results in excessively wide prediction sets. On the other hand, by the Bonferroni inequality, the intersection of the K sets ensures a coverage of at least $1 - K\alpha$, which may become meaningless as K grows. To provide a fair comparison, we obtain the conformal prediction sets at level $2\alpha/K$ using the $K=5$ different algorithms. Under this choice, the intersection achieves coverage of at least $1 - 2\alpha$

Table 2: Size ($\pm$ sd) and coverage on synthetic data ($\alpha=0.1$).

Method	Size	Coverage
LDA (Best Single)	2.488 ± 0.032	0.896
Ensemble	2.537 ± 0.032	0.898
Bonferroni	2.658 ± 0.033	0.897
SA	2.565 ± 0.033	0.919
Ada. COMA	2.490 ± 0.032	0.897

Figure 4: Weights assigned to the algorithms across iterations. After a few iterations, the adaptive- η strategy concentrates all mass on the best-performing algorithm.

(as our method). Finally, we consider EFCP and VFCP by Yang and Kuchibhotla [2025]. The EFCP method requires only the prediction sets; however, its coverage is not exactly guaranteed at the nominal level $1 - \alpha$. In contrast, VFCP guarantees coverage at the prescribed level $1 - \alpha$, but it requires an additional calibration set and thus full access to the dataset. Both methods are valid only using split conformal prediction in the batch regression setting. In general, COMA improves upon most competing aggregation methods while maintaining coverage (Table 1). The size of the resulting prediction sets is comparable to that of the EFCP method. Assumption 1 appears reasonable, as the maximum empirical counterpart of the quantity in (3) is 0.002 for both the fixed- η and adaptive- η methods.

Synthetic Data. For the classification experiment, we generate a synthetic dataset with 2000 observations as follows. We consider $p=8$ covariates sampled from an equicorrelated standard multivariate normal distribution with $\sigma_{ij} = 0.5, i \neq j$. The response $y \in \{1, \dots, 8\}$ is then drawn from a multinomial distribution with probabilities proportional to $w_j(x) \propto \exp\{x^\top \beta_j\}, j = 1, \dots, 8$, where each coefficient vector $\beta_j \in \mathbb{R}^p$ is independently sampled from a standard normal distribution. Specifically, following a burn-in phase of 1000 iterations, we re-trained four different classifiers at each step, and the weights for these algorithms are adjusted using the COMA method. The four classifiers are Linear Discriminant Analysis (LDA), Quadratic Discriminant Analysis (QDA), random forest and neural net. At each time step, split conformal inference is used to obtain a prediction set for $y^{(t)}$ and the score used is $s(x, y) = 1 - \hat{\mu}(x; y)$ where $\hat{\mu}(x; y_d)$ is the predicted probability for the label $y_d \in \mathcal{Y} = \{1, \dots, 8\}$.

We apply the COMA procedure with both adaptive and fixed

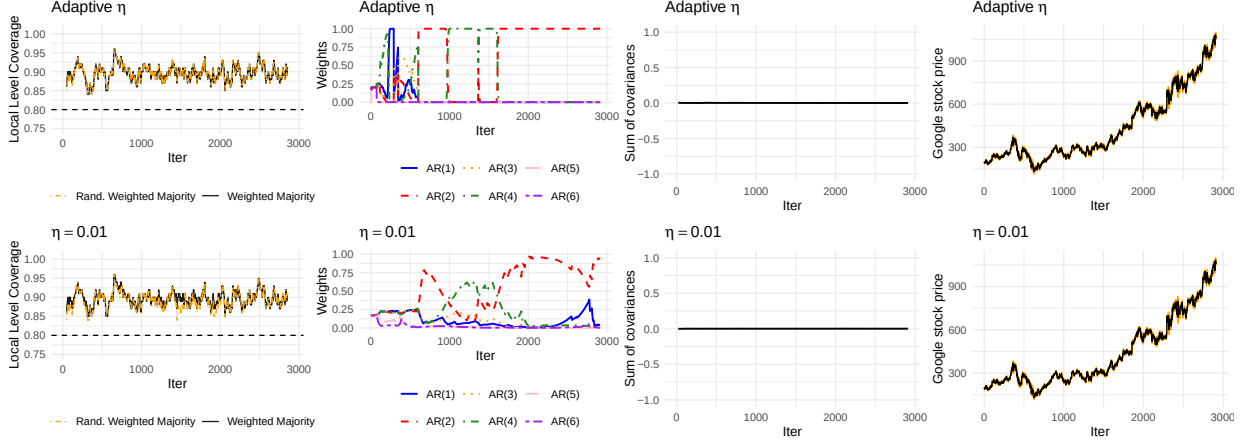


Figure 5: Local level coverage computed over a moving window of 100 data points (first column), model weights assigned across iterations (second column), running empirical estimate of the quantity $\frac{1}{T} \sum_{t=1}^T \sum_{k=1}^K (\phi_k^{(t)} - \bar{\phi}_k)(w_k^{(t)} - \bar{w}_k)$ as the number of observations increases (third column), and original stock-price series with the corresponding prediction intervals (fourth column). The black dashed lines in the first column indicate the target level $1 - 2\alpha$.

η , setting the latter to 0.1. As benchmarks, we consider the intersection of the sets obtained by training each method at miscoverage level $2\alpha/K$ and conformal prediction based on an ensemble of the base models. We also include the score averaging (SA) method of Luo and Zhou [2025], which learns a weighted combination of conformity scores with the aim of minimizing the set size. As shown in Table 2, COMA essentially matches the best base model in set size while maintaining coverage close to $1 - \alpha$. The multiplicative factor of 2 does not appear in the observed miscoverage rate, and a similar behavior arises when the intersection is constructed using the Bonferroni correction. This may be attributed to the strong dependence among the predictors, as they are all trained on the same data, whereas Bonferroni’s inequality is nearly tight under weak dependence. Figure 4 shows that the weights rapidly concentrate on the best-performing model.

6.2 EXPERIMENTS IN NON-IID SETTING

Inspired by Angelopoulos et al. [2024a], we analyze the historical series comprising the daily opening prices of Google stock from January 1, 2006, to December 31, 2014 [Nugent, 2018]. The compared models consist of six different autoregressive models incorporating temporal lags from 1 to 6, trained on the logarithmic scale of the opening price, while the forecasts are obtained on the original scale. The series exhibits nonstationary behavior, as can be observed in the last column in Figure 5. To construct the merged set, we apply the algorithm introduced in Section 5.1 (applications of the method in Section 5.2 are reported in Appendix D), setting $\alpha = 0.1$. A similar analysis is presented in Angelopoulos et al. [2024a], which considers a single autoregressive model with three lags. The parameter γ is updated as $\gamma^{(t)} = 0.1B^{(t)}$, where $B^{(t)}$ is the maximum observed

score in a trailing window of 100 iterations, and the models are re-trained at every time step. The score function corresponds to absolute residuals and prediction sets are obtained as in (11). The weights are updated either using an adaptive η or a fixed value $\eta = 0.01$.

Autoregressive models with two and four lags appear to outperform the others. In particular, with adaptive η , the weights concentrate on these models after the first few iterations, alternating between them over time. In contrast, the fixed- η strategy updates weights more conservatively and keeps aggregating models over time, while local coverage remains close to the $(1 - \alpha)$ level in both cases. The quantity in (13) remains close to zero throughout all iterations, suggesting that the assumption in (13) is reasonable in both cases. The method effectively adapts to distributional shifts while maintaining strong empirical performance.

7 CONCLUSION

Our work presents a method for aggregating conformal prediction sets produced by different algorithms. Specifically, we propose an aggregation approach in which the contribution of each individual prediction set is weighted according to its past performance. The proposed method requires access only to the prediction sets, making it particularly well suited to decentralized settings. Our approach can be viewed as a wrapper around conformal prediction, specifically designed for online settings. One limitation is that, empirically, coverage is often close to $1 - \alpha$, while the theory only guarantees at least $1 - 2\alpha$. Another limitation is that we assume the true label is observed at every round; extending the method to settings with intermittent feedback is an interesting direction for future work.

Acknowledgements

AR acknowledges funding from NSF grant IIS-2229881, and thanks Rina Foygel Barber, Ryan Tibshirani and Emmanuel Candès for useful conversations around model selection in conformal prediction.

References

- Nabil Alami, Jad Zakharia, and Souhaib Ben Taieb. Symmetric aggregation of conformity scores for efficient uncertainty sets. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 19607–19614, 2026.
- Anastasios Angelopoulos, Emmanuel Candès, and Ryan Tibshirani. Conformal PID control for time series prediction. *Advances in Neural Information Processing Systems*, 36, 2024a.
- Anastasios N. Angelopoulos and Stephen Bates. Conformal prediction: A gentle introduction. *Foundations and Trends in Machine Learning*, 16(4):494–591, 2023. ISSN 1935-8237. doi: 10.1561/22000000101.
- Anastasios N Angelopoulos, Rina Foygel Barber, and Stephen Bates. Online conformal prediction with decaying step sizes. In *International Conference on Machine Learning*, 2024b.
- Anastasios N Angelopoulos, Rina Foygel Barber, and Stephen Bates. Theoretical foundations of conformal prediction. *arXiv preprint arXiv:2411.11824*, 2024c.
- Davide Ballabio, Matteo Cassotti, Viviana Consonni, and Roberto Todeschini. QSAR aquatic toxicity. UCI Machine Learning Repository, 2014. DOI: <https://doi.org/10.24432/C5SG7H>.
- Rina Foygel Barber, Emmanuel Candès, Aaditya Ramdas, and Ryan Tibshirani. Predictive inference with the jackknife+. *The Annals of Statistics*, 49(1):486 – 507, 2021.
- Rina Foygel Barber, Emmanuel Candès, Aaditya Ramdas, and Ryan Tibshirani. Conformal prediction beyond exchangeability. *The Annals of Statistics*, 51(2):816–845, 2023.
- Mauro Bernardi and Mariangela Guidolin. The determinants of Airbnb prices in New York City: a spatial quantile regression approach. *Journal of the Royal Statistical Society Series C: Applied Statistics*, 72(1):104–143, 2023.
- Michael Bian and Rina Foygel Barber. Training-conditional coverage for distribution-free predictive inference. *Electronic Journal of Statistics*, 17(2):2044–2066, 2023.
- Andrea Campagner, Marília Barandas, Duarte Folgado, Hugo Gamboa, and Federico Cabitza. Ensemble predictors: Possibilistic combination of conformal predictors for multivariate time series classification. *IEEE Trans. Pattern Anal. Mach. Intell.*, 46(11):7205–7216, 2024. doi: 10.1109/TPAMI.2024.3388097.
- Song Chen. Beijing Multi-Site Air Quality. UCI Machine Learning Repository, 2017. DOI: <https://doi.org/10.24432/C5RK5G>.
- Victor Chernozhukov, Kaspar Wüthrich, and Zhu Yinchu. Exact and robust conformal inference methods for predictive machine learning with dependent data. In *Conference On learning theory*, pages 732–749. PMLR, 2018.
- Giovanni Cherubin. Majority vote ensembles of conformal predictors. *Machine Learning*, 108(3):475–488, 2019.
- Steven de Rooij, Tim van Erven, Peter D. Grünwald, and Wouter M. Koolen. Follow the leader if you can, hedge if you must. *Journal of Machine Learning Research*, 15 (37):1281–1316, 2014.
- Robin Dunn, Larry Wasserman, and Aaditya Ramdas. Distribution-free prediction sets for two-layer hierarchical models. *Journal of the American Statistical Association*, 118(544):2491–2502, 2023.
- Jianqing Fan, Jiawei Ge, and Debarghya Mukherjee. UTOPIA: Universally trainable optimal prediction intervals aggregation. *arXiv preprint arXiv:2306.16549*, 2023.
- Yoav Freund and Robert E Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55 (1):119–139, 1997.
- Matteo Gasparin and Aaditya Ramdas. Merging uncertainty sets via majority vote. *arXiv preprint arXiv:2401.09379*, 2024.
- Isaac Gibbs and Emmanuel Candès. Adaptive conformal inference under distribution shift. *Advances in Neural Information Processing Systems*, 34:1660–1672, 2021.
- Isaac Gibbs and Emmanuel Candès. Conformal inference for online prediction with arbitrary distribution shifts. *Journal of Machine Learning Research*, 25(162):1–36, 2024.
- Michael Harries. *SPLICE-2 Comparative Evaluation: Electricity Pricing*. Tech. report, Univ. of New South Wales, 2003.
- Pierre Humbert, Batiste Le Bars, Aurélien Bellet, and Sylvain Arlot. One-shot federated conformal prediction. In *International Conference on Machine Learning*, pages 14153–14177. PMLR, 2023.

- Abdullah Al Imran, Md Shamsur Rahim, and Tanvir Ahmed. Mining the productivity data of the garment industry. *Int. J. Bus. Intell. Data Min.*, 19(3):319–342, 2021. doi: 10.1504/ijbidm.2021.118183. URL <https://doi.org/10.1504/ijbidm.2021.118183>.
- Roger Koenker. *Quantile regression*, volume 38. Cambridge University Press, Cambridge, 2005.
- Jing Lei, Max G’Sell, Alessandro Rinaldo, Ryan Tibshirani, and Larry Wasserman. Distribution-free predictive inference for regression. *Journal of the American Statistical Association*, 113(523):1094–1111, 2018.
- N. Littlestone and M.K. Warmuth. The weighted majority algorithm. *Information and Computation*, 108(2):212–261, 1994. doi: <https://doi.org/10.1006/inco.1994.1009>.
- Rui Luo and Zhixin Zhou. Conformity score averaging for classification. In *Forty-second International Conference on Machine Learning*, 2025.
- Cam Nugent. S&P 500 stock data. <https://www.kaggle.com/datasets/camnugent/sandp500>, 2018.
- Harris Papadopoulos, Kostas Proedrou, Volodya Vovk, and Alex Gammerman. Inductive confidence machines for regression. In *13th European Conference on Machine Learning*, pages 345–356. Springer, 2002.
- Yash Patel, Eduardo Ochoa Rivera, and Ambuj Tewari. Conformal prediction for ensembles: Improving efficiency via score-based aggregation. In *Advances in Neural Information Processing Systems*, 2025.
- Aleksandr Podkopaev and Aaditya Ramdas. Distribution-free uncertainty quantification for classification under label shift. In *Uncertainty in Artificial Intelligence*, pages 844–853. PMLR, 2021.
- Aaditya Ramdas and Tudor Manole. Randomized and exchangeable improvements of Markov’s, Chebyshev’s and Chernoff’s inequalities. *Statistical Science*, 41(1):121–142, 2026.
- Michael Redmond. Communities and Crime. UCI Machine Learning Repository, 2002. DOI: <https://doi.org/10.24432/C53W3X>.
- Yaniv Romano, Evan Patterson, and Emmanuel Candès. Conformalized quantile regression. *Advances in Neural Information Processing Systems*, 32, 2019.
- Yaniv Romano, Matteo Sesia, and Emmanuel Candès. Classification with valid and adaptive coverage. *Advances in Neural Information Processing Systems*, 33:3581–3591, 2020.
- Craig Saunders, Alex Gammerman, and Volodya Vovk. Transduction with confidence and credibility. *International Joint Conference on Artificial Intelligence*, 16: 722–726, 1999.
- Aldo Solari and Vera Djordjilović. Multi-split conformal prediction. *Statistics & Probability Letters*, 184:109395, 2022. ISSN 0167-7152. doi: <https://doi.org/10.1016/j.spl.2022.109395>.
- Johan Hallberg Szabadváry and Tuwe Löfström. Beyond conformal predictors: Adaptive conformal inference with confidence predictors. *Pattern Recognition*, 170, 2025.
- Ryan Tibshirani, Rina Foygel Barber, Emmanuel Candès, and Aaditya Ramdas. Conformal prediction under covariate shift. *Advances in Neural Information Processing Systems*, 32, 2019.
- Paolo Toccaceli and Alexander Gammerman. Combination of conformal predictors for classification. *Conformal and Probabilistic Prediction and Applications*, 60:39–61, 2017.
- Vladimir G. Trunov and Vladimir V. V’yugin. Online aggregation of conformal predictive systems. In *Conformal and Probabilistic Prediction with Applications*, volume 204 of *Proceedings of Machine Learning Research*, pages 430–449. PMLR, 2023.
- Vladimir Vovk. Conditional validity of inductive conformal predictors. In *Asian conference on machine learning*, pages 475–490. PMLR, 2012.
- Vladimir Vovk. Cross-conformal predictors. *Annals of Mathematics and Artificial Intelligence*, 74:9–28, 2015.
- Vladimir Vovk, Alexander Gammerman, and Glenn Shafer. *Algorithmic Learning in a Random World*, volume 29. Springer, New York, 2005.
- Vladimir Vovk, Bin Wang, and Ruodu Wang. Admissible ways of merging p-values under arbitrary dependence. *The Annals of Statistics*, 50(1):351–375, 2022.
- Hadley Wickham. *ggplot2: Elegant Graphics for Data Analysis*. Springer-Verlag New York, New York, 2016. ISBN 978-3-319-24277-4. URL <https://ggplot2.tidyverse.org>.
- Yachong Yang and Arun Kuchibhotla. Selection and aggregation of conformal prediction sets. *Journal of the American Statistical Association*, 120(549):435–447, 2025.
- I-Cheng Yeh. Concrete Compressive Strength. UCI Machine Learning Repository, 1998. DOI: <https://doi.org/10.24432/C5PK67>.
- Margaux Zaffran, Olivier Féron, Yannig Goude, Julie Josse, and Aymeric Dieuleveut. Adaptive conformal predictions for time series. In *International Conference on Machine Learning*, pages 25834–25866. PMLR, 2022.

Conformal Online Model Aggregation (Supplementary Material)

Matteo Gasparin¹

Aaditya Ramdas²

¹Department of Statistical Sciences, University of Padova, Padova, Italy

²Department of Statistics and Data Science, Carnegie Mellon University, Pittsburgh, Pennsylvania, USA

A CONFORMAL INFERENCE AND ADAPTIVE CONFORMAL INFERENCE

In this section we briefly introduce split (or inductive) conformal prediction [Papadopoulos et al., 2002, Lei et al., 2018]. A more detailed introduction can be found in Angelopoulos and Bates [2023]. Suppose that we have a fitted prediction model $\hat{\mu}$ used to predict the value of $Y^{(t)}$ given $X^{(t)} = x^{(t)}$. In addition, let us assume that we have a calibration set $\mathcal{D}_{cal} = \{(x^{(i)}, y^{(i)})\}_{i=1}^{t-1}$ that is different from the data where we trained our prediction model. In this scenario, we introduce $s : \mathcal{X} \times \mathcal{Y} \rightarrow \mathbb{R}$ as the *conformal score* function, assessing the degree of agreement between the value y and the predictions from the model. A commonly used score function in regression analysis is the absolute residual, expressed as $s(x, y) = |y - \hat{\mu}(x)|$, where $\hat{\mu}(x)$ represents the point prediction of the model $\hat{\mu}$ using the covariates in x . We can now define the fitted quantiles of the scores in the calibration set as

$$q(\beta) := \inf \left\{ b : \frac{1}{|\mathcal{D}_{cal}|} \sum_{(x^{(i)}, y^{(i)}) \in \mathcal{D}_{cal}} \mathbb{1} \left\{ s(x^{(i)}, y^{(i)}) \leq b \right\} \geq \beta \right\},$$

with the convention $q(\beta) = +\infty$ for $\beta > 1$. The conformal prediction set for $Y^{(t)}$ based on $X^{(t)} = x^{(t)}$ with error level $\alpha \in (0, 1)$ is given by

$$\mathcal{C}^{(t)} := \mathcal{C}^{(t)}(\alpha) = \left\{ y \in \mathcal{Y} : s(x^{(t)}, y) \leq q \left(\frac{(|\mathcal{D}_{cal}| + 1)(1 - \alpha)}{|\mathcal{D}_{cal}|} \right) \right\}. \quad (16)$$

If data are iid (or exchangeable), then it is possible to prove that $\mathbb{P}(Y^{(t)} \notin \mathcal{C}^{(t)}) \leq \alpha$. The result also applies to classification problems; in this case, we only need different conformity score functions (since, for example, residual based scores are generally not appropriate). One possible solution is to use $s(x, y) = 1 - \hat{\mu}(x; y_d)$, where $\hat{\mu}(x; y_d)$ is the predicted probability for the label $y_d \in \mathcal{Y} = \{y_1, \dots, y_D\}$ given the value of x . Other proposals are described in Romano et al. [2020].

In the case of adaptive conformal inference [Gibbs and Candès, 2021], the error level varies over time, as described in (10). However, the prediction sets at each time t are obtained as in (16), where $\mathcal{D}_{cal}$ is the calibration set that contains observations before time t that were not used to train the prediction model. In the described scenario, it is not guaranteed that the level $\alpha^{(t)}$ lies in the interval $(0, 1)$. Specifically, when $\alpha^{(t)} \leq 0$, then $\mathcal{C}^{(t)}(\alpha^{(t)}) = \mathcal{Y}$, while when $\alpha^{(t)} \geq 1$, we have $\mathcal{C}^{(t)}(\alpha^{(t)}) = \emptyset$.

B ADDITIONAL DISCUSSIONS ON THE COVERAGE GUARANTEE

We now discuss the coverage guarantee in Theorem 2. In particular, we note that the expectation in the marginal coverage condition in (3) is taken over both the training data and the test point. An alternative condition is the following:

$$\mathbb{E} \left[\phi_k^{(t)} \mid \{Z^{(i)}\}_{i=1}^{t-1} \right] \leq \alpha, \text{ for all } k \in [K], \quad (17)$$

where, in this case, the miscoverage error is bounded conditionally on the “training set” $\{Z^{(i)}\}_{i=1}^{t-1}$. In practice, the model may not have been updated after each additional point, but we still call this condition *training conditional coverage*. This condition can be used to extend the results in Theorem 2.

Lemma 10. *Let $\mathcal{C}_1^{(t)}, \dots, \mathcal{C}_K^{(t)}$ be $K \geq 2$ different prediction sets for $Y^{(t)}$, let $W^{(t)}$ be a random vector in Δ_{K-1} depending only on $\{Z^{(i)}\}_{i=1}^{t-1}$, and consider the weighted majority vote set $\mathcal{C}_M^{(t)}$ in (2). If (17) holds, then $\mathbb{P}(Y^{(t)} \notin \mathcal{C}_M^{(t)}) \leq 2\alpha$.*

However, this condition does not hold in general for conformal prediction methods. A weaker type of guarantee can be obtained through PAC-type results, which involve two parameters: the target miscoverage rate α and a confidence parameter δ . In particular, under suitable conditions, the training-conditional coverage of conformal prediction methods is asymptotically (in the size of the calibration set) equal to $1 - \alpha$ with probability at least $1 - \delta$ [Vovk, 2012, Bian and Barber, 2023]. For split conformal prediction, the training-conditional miscoverage can be controlled by α plus an error term whose standard deviation decreases as the inverse square root of the calibration sample size [Angelopoulos et al., 2024c, Chapter 4]. Combining this result with the same argument used in Lemma 10, and using the fact that the weights lie in the simplex, yields a weaker guarantee in which the miscoverage is bounded by approximately 2α , up to a vanishing error term.

A different condition holds for full conformal prediction in the online iid setting [Vovk et al., 2005, Chapter 2.2], specifically

$$\mathbb{E} \left[\phi_k^{(t)} \mid \{\phi_k^{(i)}\}_{i=1}^{t-1} \right] \leq \alpha, \text{ for all } k \in [K]. \quad (18)$$

The left hand side above conditions on less information than (17). Although (18) captures the online validity of each individual full conformal predictor, it is not strong enough for our aggregation result. Indeed, this condition alone does not suffice to guarantee a miscoverage rate of at most 2α for the aggregated set.

As outlined in Theorem 7, Assumption 1 guarantees a miscoverage rate of at most 2α for the proposed solution. However, since the weights are updated based on the losses of the different algorithms, a sufficient condition which lies between (17) and (18) and guarantees a miscoverage rate of at most 2α for Algorithm 1 is

$$\mathbb{E} \left[\phi_k^{(t)} \mid \{\phi_k^{(i)}\}_{i=1}^{t-1}, \{\ell_k^{(i)}\}_{i=1, k=1}^{t-1, K} \right] \leq \alpha, \text{ for all } k \in [K].$$

However, it remains unclear whether this condition is likely to hold either in theory or in practice. For this reason, we regard Assumption 1 as the most reasonable choice in practical applications. From a theoretical point of view, this condition is difficult to verify, since the sets are based on the same underlying data and both the errors and the losses may be arbitrarily correlated. Nevertheless, it can be empirically assessed and appears to hold reasonably well in the experiments in Section 6. If the assumption is not met, Remark 5 shows that the coverage guarantee weakens gradually as the violation grows.

C ALGORITHMS FOR COMA UNDER DISTRIBUTION SHIFT

In this Section we describe the algorithms introduced in Section 5.

Algorithm 2 Decentralized COMA under distribution shift

Require: Initial learning rate $\eta^{(1)} \geq 0$ and starting quantiles $q_1, \dots, q_K$

- 1: $w_k^{(1)} \leftarrow 1/K, L_k^{(0)} \leftarrow 0, k \in [K]$
 - 2: $q_k^{(1)} = q_k, k \in [K]$
 - 3: **for** rounds $t = 1, \dots, T$ **do**
 - 4: Require $\mathcal{C}_k^{(t)}(q_k^{(t)}), k \in [K]$;
 - 5: $\mathcal{C}_M^{(t)} \leftarrow \left\{ y \in \mathcal{Y} : \sum_{k=1}^K w_k^{(t)} \mathbb{1}\{y \in \mathcal{C}_k^{(t)}\} > \frac{1}{2} \right\}$
 - 6: Receive loss $\ell_k^{(t)}$, update $L_k^{(t)} := L_k^{(t-1)} + \ell_k^{(t)}$
 - 7: Update learning rate $\eta^{(t+1)}$
 - 8: $w_k^{(t+1)} \leftarrow \exp\{-\eta^{(t+1)} L_k^{(t)}\} / \sum_{j=1}^K \exp\{-\eta^{(t+1)} L_j^{(t)}\}$
 - 9: Observe $y^{(t)}$ and errors for all experts $\phi_k^{(t)}, k \in [K]$
 - 10: Update quantiles $q_k^{(t+1)}, k \in [K]$
 - 11: **end for**
-

Algorithm 3 Centralized COMA under Distribution Shift

Require: Initial learning rate $\eta^{(1)} \geq 0$ and miscoverage rate α

```
1:  $w_k^{(1)} \leftarrow 1/K, L_k^{(0)} \leftarrow 0, k \in [K]$ 
2:  $\alpha^{(1)} = \alpha$ 
3: for rounds  $t = 1, \dots, T$  do
4:   Require  $\mathcal{C}_k^{(t)}(\alpha^{(t)})$ ,  $k \in [K]$ ;
5:    $\mathcal{C}_M^{(t)} \leftarrow \left\{ y \in \mathcal{Y} : \sum_{k=1}^K w_k^{(t)} \mathbb{1}\{y \in \mathcal{C}_k^{(t)}\} > \frac{1}{2} \right\}$ 
6:   Receive loss  $\ell_k^{(t)}$ , update  $L_k^{(t)} := L_k^{(t-1)} + \ell_k^{(t)}$ 
7:   Update learning rate  $\eta^{(t+1)}$ 
8:    $w_k^{(t+1)} \leftarrow \exp\{-\eta^{(t+1)} L_k^{(t)}\} / \sum_{j=1}^K \exp\{-\eta^{(t+1)} L_j^{(t)}\}$ 
9:   Observe  $y^{(t)}$  and miscoverage event  $\phi^{(t)} = \mathbb{1}\{y^{(t)} \notin \mathcal{C}_M^{(t)}\}$ 
10:  Update miscoverage rate  $\alpha^{(t+1)}$ 
11: end for
```

D ADDITIONAL EXPERIMENTS

D.1 REAL DATASETS (IID SETTING)

We evaluated our COMA method, described in Section 4, on multiple real datasets to assess its performance in terms of coverage and size. To construct the sets, we used split conformal prediction with absolute residuals as conformity score. We considered two scenarios with $K = 6$ base models. In the first scenario, the models used are: a linear regression, a lasso regression with regularization parameter 0.1, a ridge regression with regularization parameter 0.1, a random forest with 250 trees, a neural network with 4 nodes, and a neural network with 8 nodes. In the second case, the models are three neural nets with 4, 6, 8 nodes and three random forests with 50, 150, 250 trees.

As benchmarks, we included prediction intervals from split conformal prediction with an ensemble of the six models as base predictor (the ensemble is obtained by using a simple average), the union of the sets obtained from split conformal prediction, and the intersection of the sets obtained from split conformal prediction calibrated at level $\alpha' = 2\alpha/K$. For each dataset, we sampled 500 observations and applied the aforementioned methods using data from the 250-th observation onward. At each iteration, all models are re-trained, and the results are reported in Tables 3 and 4. In Table 5, we report the total elementwise correlation defined in (3). The empirical values are close to zero across all experiments, suggesting that Assumption 1 appears to hold. Our COMA method never produces empty sets in these experiments and is overall among the best-performing methods. The performance of COMA with fixed η and COMA with adaptive η can differ, and in some cases the fixed- η version outperforms the adaptive version. An explanation is that, after a few iterations, the adaptive- η version concentrates all the weight on the best model, whereas the fixed- η version distributes the weight across two or three algorithms. As a result, the randomized majority vote among two or three methods may align with the intersection.

Table 3: Empirical results of the different methods across multiple datasets. The base models include lasso regression, ridge regression, linear regression, random forest, and two neural networks.

Dataset	COMA Adap. η	COMA $\eta = 1$	Ensem.	Union	Bonf.	EFCP
Crime	0.447 $\pm$ 0.003	0.427 $\pm$ 0.003	0.509 $\pm$ 0.005	0.822 $\pm$ 0.011	0.610 $\pm$ 0.007	0.428 $\pm$ 0.002
[Redmond, 2002]	0.900 $\pm$ 0.019	0.888 $\pm$ 0.020	0.928 $\pm$ 0.016	0.984 $\pm$ 0.008	0.940 $\pm$ 0.015	0.900 $\pm$ 0.019
Aquatic toxicity	0.988 $\pm$ 0.006	0.956 $\pm$ 0.007	0.974 $\pm$ 0.006	1.291 $\pm$ 0.009	1.263 $\pm$ 0.009	0.909 $\pm$ 0.004
[Ballabio et al., 2014]	0.900 $\pm$ 0.019	0.896 $\pm$ 0.019	0.940 $\pm$ 0.015	0.972 $\pm$ 0.011	0.960 $\pm$ 0.012	0.904 $\pm$ 0.019
Diamonds	0.810 $\pm$ 0.003	0.782 $\pm$ 0.004	0.887 $\pm$ 0.004	1.252 $\pm$ 0.013	0.928 $\pm$ 0.008	0.790 $\pm$ 0.002
[Wickham, 2016]	0.892 $\pm$ 0.020	0.884 $\pm$ 0.020	0.908 $\pm$ 0.018	0.968 $\pm$ 0.011	0.936 $\pm$ 0.016	0.888 $\pm$ 0.020
Concrete	22.415 $\pm$ 0.251	24.744 $\pm$ 0.174	28.529 $\pm$ 0.146	42.885 $\pm$ 0.146	28.384 $\pm$ 0.363	23.437 $\pm$ 0.106
[Yeh, 1998]	0.920 $\pm$ 0.042	0.952 $\pm$ 0.014	0.920 $\pm$ 0.017	0.992 $\pm$ 0.006	0.940 $\pm$ 0.015	0.956 $\pm$ 0.013
Productivity	0.455 $\pm$ 0.002	0.453 $\pm$ 0.003	0.546 $\pm$ 0.003	0.681 $\pm$ 0.004	0.598 $\pm$ 0.005	0.454 $\pm$ 0.005
[Imran et al., 2021]	0.900 $\pm$ 0.019	0.888 $\pm$ 0.020	0.908 $\pm$ 0.018	0.960 $\pm$ 0.012	0.940 $\pm$ 0.015	0.900 $\pm$ 0.019

Table 4: Empirical results of the different methods across multiple datasets. The base models include three random forests and three neural networks with different tuning parameters.

Dataset	COMA Adap. η	COMA $\eta = 1$	Ensem.	Union	Bonf.	EFCP
Crime [Redmond, 2002]	0.443 $\pm$ 0.003 0.916 $\pm$ 0.018	0.413 $\pm$ 0.004 0.896 $\pm$ 0.019	0.482 $\pm$ 0.005 0.932 $\pm$ 0.016	0.614 $\pm$ 0.005 0.980 $\pm$ 0.009	0.606 $\pm$ 0.005 0.980 $\pm$ 0.009	0.426 $\pm$ 0.006 0.928 $\pm$ 0.016
Aquatic toxicity [Ballabio et al., 2014]	0.905 $\pm$ 0.009 0.912 $\pm$ 0.018	0.875 $\pm$ 0.008 0.904 $\pm$ 0.019	0.980 $\pm$ 0.007 0.936 $\pm$ 0.016	1.262 $\pm$ 0.010 0.992 $\pm$ 0.006	1.237 $\pm$ 0.010 0.956 $\pm$ 0.013	0.901 $\pm$ 0.003 0.956 $\pm$ 0.013
Diamonds [Wickham, 2016]	0.791 $\pm$ 0.003 0.892 $\pm$ 0.020	0.769 $\pm$ 0.004 0.888 $\pm$ 0.020	0.834 $\pm$ 0.004 0.932 $\pm$ 0.016	1.060 $\pm$ 0.008 0.988 $\pm$ 0.007	0.900 $\pm$ 0.006 0.940 $\pm$ 0.015	0.780 $\pm$ 0.002 0.888 $\pm$ 0.020
Concrete [Yeh, 1998]	22.274 $\pm$ 0.216 0.920 $\pm$ 0.042	23.867 $\pm$ 0.157 0.940 $\pm$ 0.015	25.007 $\pm$ 0.160 0.928 $\pm$ 0.016	35.418 $\pm$ 0.332 0.996 $\pm$ 0.004	26.228 $\pm$ 0.371 0.920 $\pm$ 0.018	22.249 $\pm$ 0.089 0.920 $\pm$ 0.017
Productivity [Imran et al., 2021]	0.431 $\pm$ 0.003 0.896 $\pm$ 0.019	0.416 $\pm$ 0.003 0.888 $\pm$ 0.020	0.501 $\pm$ 0.003 0.912 $\pm$ 0.018	0.633 $\pm$ 0.004 0.972 $\pm$ 0.010	0.569 $\pm$ 0.004 0.940 $\pm$ 0.015	0.433 $\pm$ 0.002 0.920 $\pm$ 0.018

Table 5: Empirical total elementwise correlation, defined in (3), observed in the different datasets. In the first part, the base models include lasso regression, ridge regression, linear regression, random forest, and two neural networks. In the second part, the base models consist of three random forests and three neural networks with different tuning parameters.

	Crime	Aquatic toxicity	Diamonds	Concrete	Productivity	Crime	Aquatic toxicity	Diamonds	Concrete	Productivity
COMA (Adap. η)	−0.001	0.000	0.000	0.003	−0.001	0.006	−0.010	0.001	−0.006	−0.001
COMA ($\eta=1$)	0.003	0.001	0.003	−0.001	−0.006	0.002	−0.007	0.003	−0.012	−0.002

D.2 EXPERIMENTS IN NON-IID SETTING

We now evaluate our COMA procedure as a wrapper for quantile tracking and ACI through some experiments on real and simulated datasets using the methods proposed in Sections 5.1 and 5.2.

D.2.1 ELEC2 Dataset

We now use the algorithm described in Section 5.1 in a real-world application. We define $K = 3$ different regression algorithms, and we employ quantile tracking with a decaying step size $\gamma^{(t)} \propto t^{-1/2-\epsilon}$. The score function corresponds to absolute residuals. In particular, we use the ELEC2 [Harries, 2003] dataset that monitors electricity consumption and pricing in the states of New South Wales and Victoria in Australia, with data recorded every 30 minutes over a 2.5-year period from 1996 to 1999. For our experiment, we utilize four covariates: `nswprice` and `vicprice`, representing the electricity prices in each respective state, and `nswdemand` and `vicedemand`, denoting the usage demand in each state. The response variable is `transfer`, indicating the quantity of electricity transferred between the two states. We narrow our focus to a subset of the data, retaining only observations within the time range of 9 am to 12 pm to mitigate daily fluctuation effects. The models used correspond to three distinct linear models based on different covariates. The first model relies on the entire set of covariates at time t , the second model employs $y^{(t-1)}$ and $y^{(t-2)}$ as covariates (it corresponds to an AR(2)), while the third model supplements the latter with the covariate `nswprice`.

The coverage performance of the methods is compared using the local level coverage on 200 data points. The local coverage is also used to measure the performance of our merging procedure. In particular, we compare the use of an adaptive learning parameter and a small fixed $\eta = 0.1$. As can be seen in Figure 6, the three base models and the adaptive COMA procedure obtain local level coverage that oscillates around the prespecified level $1 - \alpha$. This is because the weights concentrate on one model over time; in particular, the chosen model is not always the same, as can be seen in the last column in Figure 6. The procedure proves appealing in this setting: when a clear winner emerges, it concentrates the weights on the best model

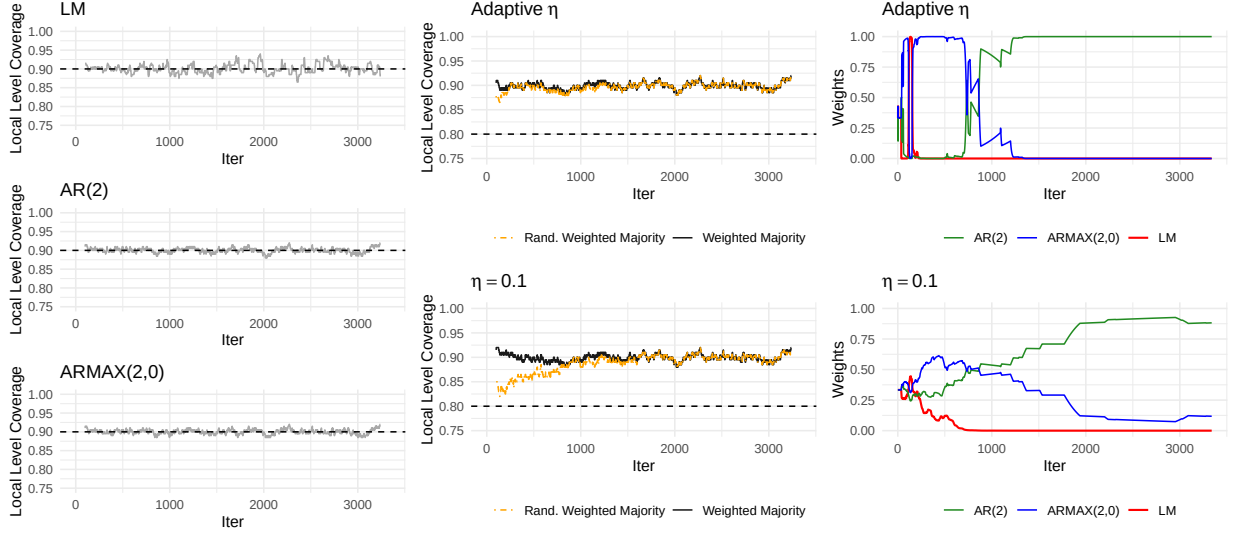


Figure 6: First column: Local level coverage for the quantile tracking using the three regression models. The dashed line represents the level $1 - \alpha$. Second column: Local level coverage for the COMA method with adaptive η or $\eta = 0.1$. The dashed line represents the level $1 - 2\alpha$. Third column: Weights obtained for the various models for the decentralized COMA algorithm applied in ELEC2 dataset. The strategy with adaptive $\eta^{(t)}$ is more aggressive than the one with $\eta^{(t)} = \eta$ in assigning weights and it performs model selection in most of the iterations.

Table 6: Cumulative loss ($L^{(T)}$) and empirical coverage obtained by the various methods. The losses of the proposed methods are better (or similar) than the cumulative loss obtained by the best algorithm. The column $\eta = 0$ represents a baseline where weights are not updated.

	LM	AR	ARMAX	Adapt. η	Adapt. η + Rand.	$\eta = 0.1$	$\eta = 0.1$ + Rand.	$\eta = 0$
$L^{(T)}$	1288.76	787.04	807.16	788.20	780.34	789.51	745.53	808.24
Coverage	0.900	0.902	0.902	0.900	0.897	0.901	0.889	0.905

while, in the absence of dominance, it allocates them across the most appropriate models while maintaining coverage.

In Table 6, we compare the cumulative loss and the empirical coverage obtained by the various methods proposed with and without the use of randomization. In addition, the case $\eta = 0$ corresponds to the majority vote with uniform weights. As can be seen, the cumulative loss of our proposed methods (with fixed and adaptive η) is lower than that of the standard majority vote approach in all cases. The best result in terms of cumulative loss, in this case, is obtained by fixing $\eta = 0.1$ and adding randomization. This is due to the fact that during some iterations the two best models have a similar weight and their sum is close to one. In this situation, the majority vote reduces to the intersection of the two sets most of the time. As an example, suppose that $w^{(t)} \approx (0.6, 0.4, 0)$ and $u^{(t)} \approx 1/2$, then the threshold is $3/4$ and in this case the majority vote set contains only points belonging to both of the first two sets, so it reduces to the intersection of $\mathcal{C}_1^{(t)}$ and $\mathcal{C}_2^{(t)}$. Indeed, during the first iterations, the local level coverage for COMA with fixed η and randomization is observed to be closer to $1 - 2\alpha$. The cumulative loss of the methods using randomization is smaller than the cumulative loss incurred by the best method, which in this case is AR(2). In particular, the factor of 2 in (7) does not appear. The empirical coverage is approximately 0.9 in all scenarios. The maximum total elementwise correlation between the weight and error vectors defined in (13) is 0.02, suggesting that the assumption of negative total elementwise empirical correlation appears to hold. The proportion of empty sets produced by both methods is below 1%.

D.2.2 ELEC2 Dataset — Centralized Setting

We study the empirical performance of the *centralized* method described in Section 5.2. We applied the procedure to the ELEC2 dataset, fixing the target coverage level at 0.9 and adopting the loss function $\arctan(\ell^{(t)})$. The use of a bounded

loss function is necessary, since ACI can output $\mathbb{R}$ as an interval. We consider both the adaptive η strategy and a fixed $\eta = 0.1$, using the three regression algorithms described in Section D.2.1. In addition, we apply ACI with a regression model obtained by ensembling the three algorithms by a simple average. The learning rate of the ACI algorithm is set to 0.05 for all methods, and we use local level coverage to compare their coverage performance.

The results, reported in Figure 7, show that the local coverage oscillates around the target level $1 - \alpha$ for all three methods. The methods yield comparable results in terms of empirical size and in the proportion of cases where the sets coincide with $\mathbb{R}$ (i.e., when $\alpha^{(t)} \leq 0$). However, our strategy with adaptive η achieves the best performance in terms of set size (Table 7), whereas the strategy with fixed η has the smallest proportion of cases where the sets coincide with $\mathbb{R}$.

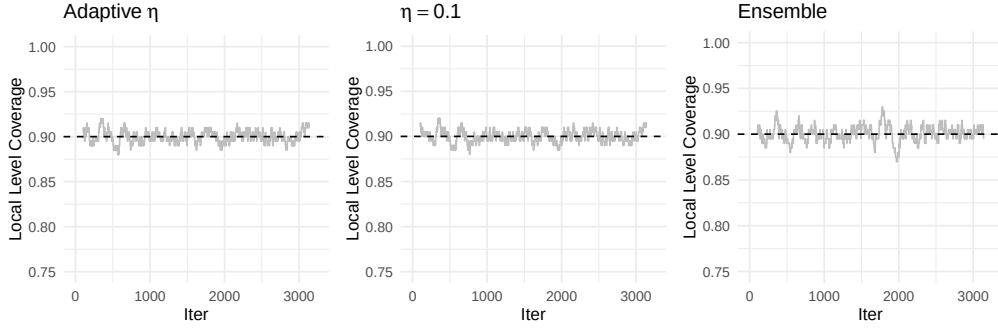


Figure 7: Local level coverage for *direct* ACI (Section 5.2) for COMA with adaptive η and $\eta = 0.1$ applied to ELEC2 dataset. The third column represents ACI directly applied to an ensemble of the three base models. The dashed lines represent the target level $1 - \alpha$.

Table 7: Empirical size and percentage of times the sets equal $\mathcal{Y}$. The strategy with adaptive η achieves the best performance in terms of empirical size, while the proportion of cases where the methods produce sets of infinite size is negligible.

	Adaptive η	$\eta = 0.1$	Ensemble
Empirical size	0.230	0.234	0.238
% set= $\mathcal{Y}$	0.926%	0.556%	1.575%

D.2.3 Additional Simulations on Quantile Tracking

We investigate the quantile tracking method in conjunction with “decentralized” COMA with adaptive η (Section 5.1). We conducted a case study in which changes in the marginal distribution also impact the weights of the two algorithms. In this case, we use simulated data and we let the distribution vary during the iterations as described below. We suppose that we have two covariates, x_1 and x_2 , and the data are generated in blocks, with only one of the two covariates influencing the response at a time. Specifically, at each time t we have $y^{(t)} = 2x^{(t)} + \varepsilon^{(t)}$ where

$$x^{(t)} = \begin{cases} x_1^{(t)}, & t \in \{[0, 50) \cup [100, 150) \cup [250, 350) \cup [450, 550) \cup \dots \cup [1250, 1350)\} \\ x_2^{(t)}, & t \in \{[50, 100) \cup [150, 250) \cup [350, 450) \cup \dots \cup [1350, 1450)\} \end{cases}$$

and $x_1^{(t)}, x_2^{(t)}, \varepsilon^{(t)}$ are generated independently from a standard normal distribution. Two standard linear models are used, one using x_1 as a regressor and the other using x_2 . Prediction intervals are generated using the quantile tracking method with the absolute residual as the score function. The models are re-trained at every time step using the most recent 100 observations. The parameter γ is set to 1 and the entire procedure is repeated 1000 times.

In the case where only two models are utilized, the majority vote (without randomization) is reduced to selecting the model with the higher weight (it performs model selection by definition). From Figure 8, the COMA method alternates the weights based on changes in the marginal distribution. In other words, the method can adapt to the changes and perform model selection. In particular, within each time window, the selected model switches between the two models around the midpoint of the window. The less aggressive behavior in assigning weights is due to the fact that $\eta^{(t)}$ decreases over time. The empirical miscoverage over the 1000 replications of our COMA procedure is 0.113, while the miscoverage rates for the two

linear models are respectively 0.101 and 0.102. The maximum empirical correlation between the weights and the errors across the 1000 replications is 0.167.

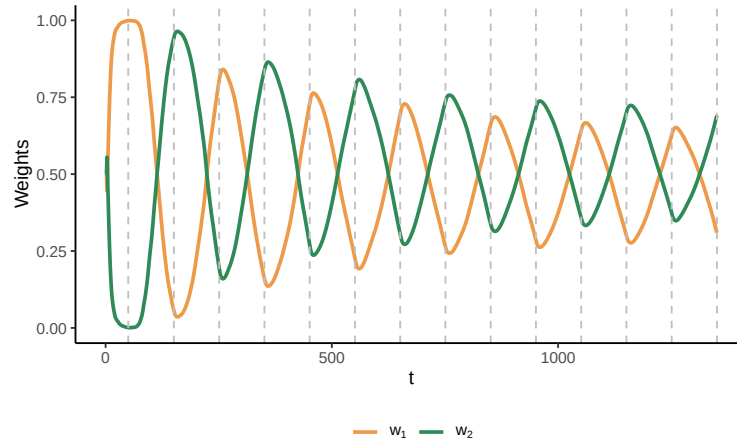


Figure 8: Average weights across 1000 replications of the linear models, using either x_1 or x_2 as the regressor. The dotted lines indicate the time points at which the distribution changes.

D.2.4 Classification Under Distribution Shift

We consider a classification example in a decentralized setting, conducted under a non-iid scenario. The dataset consists of hourly air quality measurements collected in the city of Changping [Chen, 2017]. The response variable corresponds to the levels of PM10 concentration, discretized into five classes of equal cardinality, while the covariates comprise meteorological indicators and the concentrations of other atmospheric pollutants. The models, available in the R package `mlr`, are a random forest with 150 trees, a neural network with 4 nodes, a neural network with 6 nodes, and Linear Discriminant Analysis (LDA). For this experiment, we employed the ACI method with the parameter γ set to 0.01. Since we are dealing with a classification problem, the loss is bounded and therefore cannot take infinite values (even if $\alpha^{(t)} < 0$). The significance level α is set to 0.1.

Also in this case, where the loss takes only integer values, the sets obtained via our COMA procedure attain a loss that is comparable to, and in some instances even better than, that of the best model. Moreover, the empirical coverage remains close to the nominal level $1 - \alpha$ (see Table 8). The quantity in (13) remains near zero across all iterations, with a maximum of 0.012 for COMA with adaptive η and 0.006 for COMA with fixed η .

Table 8: Cumulative loss ($L^{(T)}$) and empirical coverage obtained by the various methods. The column $\eta = 0$ represents a baseline where weights are not updated.

	RF	NN (4)	NN (6)	LDA	Adapt. η	Adapt. η + Rand.	$\eta = 0.01$	$\eta = 0.01$ + Rand.	$\eta = 0$
$L^{(T)}$	1962	2355	2267	2414	1964	1959	1967	1945	2016
Coverage	0.895	0.894	0.895	0.894	0.896	0.895	0.896	0.890	0.891

D.2.5 Amazon stock prices

We replicate the analysis described in Section 6.2 using the time series of Amazon stock prices from January 1, 2006, to December 31, 2014. The results, reported in Figure 9, show that the empirical coverage remains close to the level $1 - \alpha$. In addition, the weights fluctuate across iterations, illustrating that different methods yield varying results in different periods. The total covariance appears to be stable around zero across all iterations.

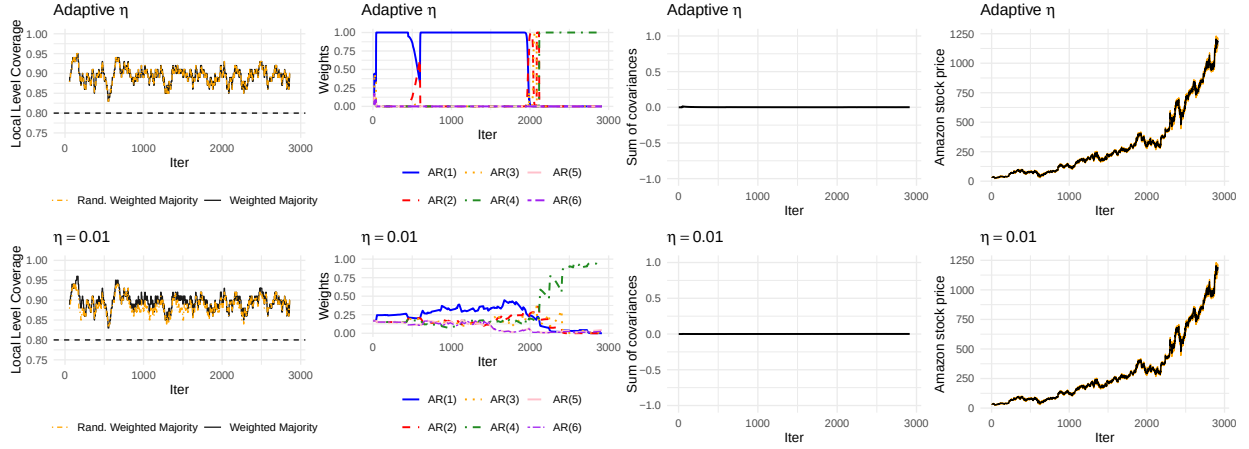


Figure 9: Local level coverage computed over a moving window of 100 data points (first column), model weights assigned across iterations (second column), running empirical estimate of the quantity $\frac{1}{T} \sum_{t=1}^T \sum_{k=1}^K (\phi_k^{(t)} - \bar{\phi}_k)(w_k^{(t)} - \bar{w}_k)$ as the number of observations increases (third column), and original stock-price series with the corresponding prediction intervals (fourth column). The black dashed lines in the first column indicate the target level $1 - 2\alpha$.

D.2.6 Additional Simulations on the Negative Correlation Assumption

In this section, we investigate the assumption of total negative correlation in Theorem 2 via a simulation study. We study the specific case introduced in Chernozhukov et al. [2018], where the data are simulated as follows: $y^{(t)} = \beta^T x^{(t)} + \varepsilon^{(t)}$, $t = 1, \dots, 200$, where $x^{(t)}$ is distributed as $\mathcal{N}(0, I_{100})$. The serial dependence is induced by generating the error as follows: $\varepsilon^{(t)} = \rho \varepsilon^{(t-1)} + \xi^{(t)}$, $\xi^{(t)} \stackrel{iid}{\sim} \mathcal{N}(0, 1 - \rho^2)$, where $\rho = 0.1$ and $\varepsilon^{(0)} = 0$. The coefficients are $\beta = (4/\sqrt{5}, 4/\sqrt{5}, 4/\sqrt{5}, 4/\sqrt{5}, 4/\sqrt{5}, 0, \dots, 0)$. For each time step $t = 100, \dots, 200$, the model is re-trained and a prediction interval for $y^{(t)}$ based on $x^{(t)}$ is constructed using Algorithm 1 in Chernozhukov et al. [2018] with $\alpha = 0.1$ and absolute residuals as score function. In particular, the algorithm chosen in this case is Lasso with 4 different penalty parameters. Under suitable conditions, Chernozhukov et al. [2018] prove that the coverage is approximately valid.

The procedure is repeated 1000 times, and we report the empirical average over the number of replications. As can be seen in Figure 10, the empirical counterpart of $\mathbb{E}[\sum_{k=1}^K (\phi_k^{(t)} - \mathbb{E}[\phi_k^{(t)}])(W_k^{(t)} - \mathbb{E}[W_k^{(t)}])]$, obtained as the empirical mean of the quantities across the replications, is around 0 for all iterations $t = 100, \dots, 200$. Although it is positive in some cases, it is only slightly above zero. The weights for each replication are obtained using the COMA method with an adaptive η , and on average they concentrate their mass in the penalty parameters λ_2 and λ_3 . The coverage level for the COMA method is between levels $1 - 2\alpha$ and $1 - \alpha$.

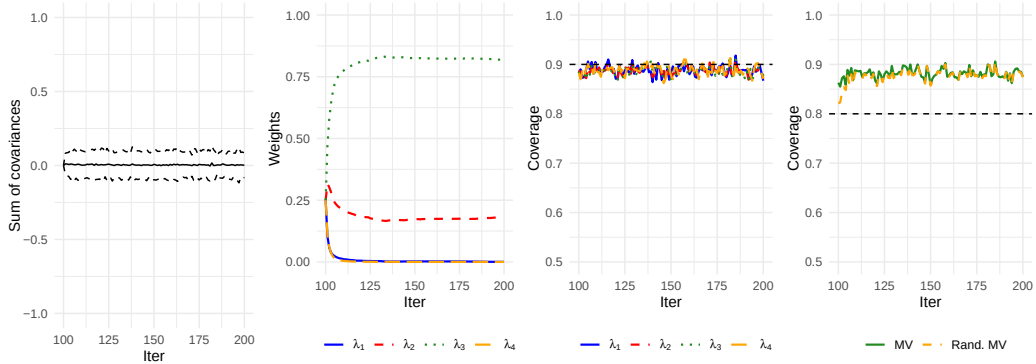


Figure 10: In the first plot, the empirical counterpart of the quantity $\mathbb{E}[\sum_{k=1}^K (\phi_k^{(t)} - \mathbb{E}[\phi_k^{(t)}])(W_k^{(t)} - \mathbb{E}[W_k^{(t)}])]$ is reported ($\pm$ sd), for all $t = 100, \dots, 200$. In the second plot, the weights assigned to the various algorithms are reported. The last two plots show the coverage of the different algorithms (third column) and the coverage of the COMA method with and without randomization (fourth column). The black dashed lines represent the levels $1 - \alpha$ and $1 - 2\alpha$.

E PROOF OF THE RESULTS

Proof of Theorem 2. By direct calculation, we can write

$$\mathbb{P}\left(Y^{(t)} \notin \mathcal{C}_M^{(t)}\right) = \mathbb{P}\left(\sum_{k=1}^K W_k^{(t)} \phi_k^{(t)} \geq \frac{1}{2} - U^{(t)}/2\right) \stackrel{(i)}{\leq} 2\mathbb{E}\left[\sum_{k=1}^K W_k^{(t)} \phi_k^{(t)}\right]$$

where (i) is valid due to the Uniformly-randomized Markov Inequality described in Ramdas and Manole [2026].

The first part of the theorem is a direct consequence of Hölder inequality. Indeed, due to the fact that $\|W^{(t)}\|_1 = 1$, we notice that

$$\mathbb{E}\left[\sum_{k=1}^K \phi_k^{(t)} W_k^{(t)}\right] \leq \mathbb{E}\left[\|\phi^{(t)}\|_\infty \|W^{(t)}\|_1\right] = \mathbb{E}\left[\|\phi^{(t)}\|_\infty\right],$$

where $\phi^{(t)} = (\phi_1^{(t)}, \dots, \phi_K^{(t)})$. In the case of negative total elementwise correlation it holds that $\sum_{k=1}^K \mathbb{E}[W_k^{(t)} \phi_k^{(t)}] \leq \sum_{k=1}^K \mathbb{E}[W_k^{(t)}] \mathbb{E}[\phi_k^{(t)}]$ and under marginal coverage, we have that

$$\mathbb{P}\left(Y^{(t)} \notin \mathcal{C}_M^{(t)}\right) \leq 2 \sum_{k=1}^K \mathbb{E}\left[W_k^{(t)} \phi_k^{(t)}\right] \leq 2 \sum_{k=1}^K \mathbb{E}[W_k^{(t)}] \mathbb{E}[\phi_k^{(t)}] \leq 2\alpha \sum_{k=1}^K \mathbb{E}[W_k^{(t)}] = 2\alpha.$$

Since the target is contained with probability at least $1 - 2\alpha$, it follows that the set is non-empty with at least the same probability. $\square$

Proof of Lemma 6. We start with the discrete case $\mathcal{Y} = \{y_1, \dots, y_D\}$. In particular, let $\mathcal{C}_1^{(t)}, \dots, \mathcal{C}_K^{(t)}$ be K different sets and let $w^{(t)}$ be any realization of $W^{(t)}$; then, we have that

$$\begin{aligned} \ell_M^{(t)} &= \sum_{d=1}^D \mathbb{1}\left\{\sum_{k=1}^K w_k^{(t)} \mathbb{1}\{y_d \in \mathcal{C}_k^{(t)}\} > \frac{1}{2} + \frac{u^{(t)}}{2}\right\} \leq \sum_{d=1}^D \mathbb{1}\left\{\sum_{k=1}^K w_k^{(t)} \mathbb{1}\{y_d \in \mathcal{C}_k^{(t)}\} > \frac{1}{2}\right\} \\ &\stackrel{(i)}{\leq} \sum_{d=1}^D 2 \sum_{k=1}^K w_k^{(t)} \mathbb{1}\{y_d \in \mathcal{C}_k^{(t)}\} = 2 \sum_{k=1}^K w_k^{(t)} \sum_{d=1}^D \mathbb{1}\{y_d \in \mathcal{C}_k^{(t)}\} = 2 \sum_{k=1}^K w_k^{(t)} \ell(\mathcal{C}_k^{(t)}), \end{aligned}$$

where (i) holds due to the fact that $\mathbb{1}\{x > 1\} \leq x$, if $x \geq 0$. The proof in the case $\mathcal{Y} \subseteq \mathbb{R}$ is identical; it replaces the summation over d by an integral.

For the second part, we start with the discrete case $\mathcal{Y} = \{y_1, \dots, y_D\}$. Fix $y_d \in \mathcal{Y}$, then

$$\begin{aligned} \mathbb{P}\left(y_d \in \mathcal{C}_M^{(t)} \mid W^{(t)}, \mathcal{C}_1^{(t)}, \dots, \mathcal{C}_K^{(t)}\right) &= \mathbb{P}\left(\sum_{k=1}^K W_k^{(t)} \mathbb{1}\{y_d \in \mathcal{C}_k^{(t)}\} > \frac{1 + U^{(t)}}{2} \mid W^{(t)}, \mathcal{C}_1^{(t)}, \dots, \mathcal{C}_K^{(t)}\right) \\ &= \mathbb{P}\left(U^{(t)} < 2 \sum_{k=1}^K W_k^{(t)} \mathbb{1}\{y_d \in \mathcal{C}_k^{(t)}\} - 1 \mid W^{(t)}, \mathcal{C}_1^{(t)}, \dots, \mathcal{C}_K^{(t)}\right) \\ &= \left(2 \sum_{k=1}^K W_k^{(t)} \mathbb{1}\{y_d \in \mathcal{C}_k^{(t)}\} - 1\right)_+ \leq \sum_{k=1}^K W_k^{(t)} \mathbb{1}\{y_d \in \mathcal{C}_k^{(t)}\}, \end{aligned}$$

where $(x)_+ = \max\{x, 0\}$. Integrating over $\mathcal{Y}$, we obtain

$$\begin{aligned} \mathbb{E}\left[\ell_M^{(t)}\right] &= \mathbb{E}\left[\sum_{d=1}^D \mathbb{1}\left\{\sum_{k=1}^K W_k^{(t)} \mathbb{1}\{y_d \in \mathcal{C}_k^{(t)}\} > \frac{1 + U^{(t)}}{2}\right\}\right] \\ &= \mathbb{E}\left[\sum_{d=1}^D \mathbb{E}\left[\mathbb{1}\left\{\sum_{k=1}^K W_k^{(t)} \mathbb{1}\{y_d \in \mathcal{C}_k^{(t)}\} > \frac{1 + U^{(t)}}{2}\right\} \mid W^{(t)}, \mathcal{C}_1^{(t)}, \dots, \mathcal{C}_K^{(t)}\right]\right] \\ &\leq \mathbb{E}\left[\sum_{d=1}^D \sum_{k=1}^K W_k^{(t)} \mathbb{1}\{y_d \in \mathcal{C}_k^{(t)}\}\right] = \mathbb{E}\left[\sum_{k=1}^K W_k^{(t)} \ell_k^{(t)}\right]. \end{aligned}$$

The proof in the case $\mathcal{Y} \subseteq \mathbb{R}$ is analogous, due to the Fubini-Tonelli Theorem. $\square$

Proof of Theorem 7. The first part of the theorem is just a consequence of Theorem 2 applied to Algorithm 1. From Lemma 6 we have $\ell_M^{(t)} \leq 2h^{(t)}$. This implies that $L_M^{(T)} = \ell_M^{(1)} + \dots + \ell_M^{(T)} \leq 2H^{(T)}$. Next, by Theorem 8 in de Rooij et al. [2014], we have

$$H^{(T)} \leq L_*^{(T)} + 2 \left(V^{(T)} \ln K \frac{(L_+^{(T)} - L_*^{(T)})(L_*^{(T)} - L_-^{(T)})}{L_+^{(T)} - L_-^{(T)}} \right)^{1/2} + V^{(T)} \left(\frac{16}{3} \ln K + 2 \right). \quad (19)$$

Combining the earlier two statements yields the desired claim. If the loss range is bounded, then $v^{(t)} \leq D$, and using Corollary 9 in de Rooij et al. [2014] we have

$$\begin{aligned} L_M^{(T)} &\leq 2L_*^{(T)} + 4 \left(\sum_{t=1}^T (v^{(t)})^2 \ln K \right)^{1/2} + 2V^{(T)} \left(\frac{4}{3} \ln K + 2 \right) \\ &\leq 2L_*^{(T)} + 4(TD^2 \ln K)^{1/2} + 2D \left(\frac{4}{3} \ln K + 2 \right). \end{aligned}$$

Dividing both sides by T gives the result. $\square$

Proof of Proposition 9. We remark that, if $u^{(t)} = 0$, then $y^{(t)}$ is not contained in $\mathcal{C}_M^{(t)}$ if and only if $\sum_{k=1}^K w_k^{(t)} \mathbb{1}\{y^{(t)} \notin \mathcal{C}_k^{(t)}(\alpha_k^{(t)})\} \geq 1/2$. Then, since $\phi_k^{(t)} = \mathbb{1}\{y^{(t)} \notin \mathcal{C}_k^{(t)}(\alpha_k^{(t)})\}$, we can write

$$\begin{aligned} \sum_{t=1}^T \mathbb{1}\{y^{(t)} \notin \mathcal{C}_M^{(t)}\} &= \sum_{t=1}^T \mathbb{1} \left\{ \sum_{k=1}^K w_k^{(t)} \mathbb{1}\{y^{(t)} \notin \mathcal{C}_k^{(t)}(\alpha_k^{(t)})\} \geq \frac{1}{2} \right\} \\ &\stackrel{(i)}{\leq} \sum_{t=1}^T 2 \sum_{k=1}^K w_k^{(t)} \mathbb{1}\{y^{(t)} \notin \mathcal{C}_k^{(t)}(\alpha_k^{(t)})\} = 2 \sum_{k=1}^K \sum_{t=1}^T w_k^{(t)} \phi_k^{(t)} \\ &\stackrel{(ii)}{\leq} 2 \sum_{k=1}^K \frac{1}{T} \left(\sum_{t=1}^T w_k^{(t)} \right) \left(\sum_{t=1}^T \phi_k^{(t)} \right) \\ &\stackrel{(iii)}{\leq} 2 \sum_{k=1}^K \frac{1}{T} \left(\sum_{t=1}^T w_k^{(t)} \right) \left(T\alpha + \frac{1 - \alpha + \gamma}{\gamma} \right) \\ &= 2 \left(\alpha + \frac{1 - \alpha + \gamma}{T\gamma} \right) \sum_{k=1}^K \sum_{t=1}^T w_k^{(t)} \\ &= 2T\alpha + 2 \frac{1 - \alpha + \gamma}{\gamma}, \end{aligned}$$

where (i) holds due to the fact that $\mathbb{1}\{x \geq 1\} \leq x$, if $x \geq 0$ and (ii) is due to assumption (13), in fact if the assumption holds, then $T \sum_{k=1}^K \sum_{t=1}^T w_k^{(t)} \phi_k^{(t)} - \sum_{k=1}^K (\sum_{t=1}^T \phi_k^{(t)}) (\sum_{t=1}^T w_k^{(t)}) \leq 0$. Proposition 4.1 in Gibbs and Candès [2021] along with $\alpha \in (0, 1/2]$ implies (iii). Dividing both sides by T gives the result. $\square$

Proof of Lemma 10. Following the same steps as in Theorem 2, if the assumption holds, then

$$\mathbb{P} \left(Y^{(t)} \notin \mathcal{C}_M^{(t)} \right) \leq 2 \sum_{k=1}^K \mathbb{E} \left[W_k^{(t)} \phi_k^{(t)} \right] = 2 \sum_{k=1}^K \mathbb{E} \left[W_k^{(t)} \mathbb{E} \left[\phi_k^{(t)} \mid \{Z^{(i)}\}_{i=1}^{t-1} \right] \right] \leq 2\alpha.$$

$\square$