Calibrated Multi-Level Quantile Forecasting

Tiffany Ding Isaac Gibbs Ryan J. Tibshirani

University of California, Berkeley

Abstract

We develop an online method that guarantees calibration of quantile forecasts at multiple quantile levels simultaneously. In this work, a sequence of quantile forecasts is said to be *calibrated* provided that its α-level predictions are greater than or equal to the target value at an α fraction of time steps, for each level α. Our procedure, called the *multi-level quantile tracker* (MultiQT), is lightweight and wraps around any point or quantile forecaster to produce adjusted quantile forecasts that are guaranteed to be calibrated, even against adversarial distribution shifts. Critically, it does so while ensuring that the quantiles remain ordered, e.g., the 0.5-level quantile forecast will never be larger than the 0.6-level forecast. Moreover, the method has a no-regret guarantee, implying it will not degrade the performance of the existing forecaster (asymptotically), with respect to the quantile loss. In our experiments, we find that MultiQT significantly improves the calibration of real forecasters in epidemic and energy forecasting problems, while leaving the quantile loss largely unchanged or slightly improved.

1 Introduction

Probabilistic forecasts are commonly conveyed via quantiles. An α-level quantile forecast attempts to predict the value below which some unknown target outcome y_t falls with probability α. Consider a forecaster that, at each time t, outputs a vector of quantile forecasts

$$q_t = (q_t^{\alpha_1}, q_t^{\alpha_2}, \ldots, q_t^{\alpha_m}),$$

for prespecified quantile levels $\mathcal{A} = \{\alpha_1, \alpha_2, \ldots, \alpha_m\}$, where $0 < \alpha_1 < \alpha_2 < \cdots < \alpha_m$. Forecasts of this type inform decision making in a wide range of applications, such as public health (Doms et al., 2018; Lutz et al., 2019), inventory management (Cao and Shen, 2019), and energy grid operation (Hong and Fan, 2016). When decisions are made on the basis of forecasts that are *calibrated*, this can lead to a reliability guarantee. For example, if a retailer has access to a sequence of calibrated 0.95-level quantile forecasts of weekly demand, and they ensure their inventory level meets these demand forecasts, then this guarantees that they run out of stock at most 5% of weeks.

Although a single ($m = 1$) quantile is sometimes sufficient for decision making, this is not true in general. When there are multiple downstream users, each with different risk tolerances and uses for the forecasts, it is often more useful to provide forecasts at multiple ($m \geq 2$) quantile levels. In this work, we seek to produce multi-level quantile forecasts that satisfy the following two useful properties:

1. *Calibration.* For any sequence of target values $y_1, y_2, \ldots$, including sequences chosen adversarially, the long-run coverage of the α-level quantile forecasts should converge to α for each $\alpha \in \mathcal{A}$. That is, if we

define $\text{cov}_t^\alpha = \mathbb{1}\{y_t \le q_t^\alpha\}$, then we want

$$\lim_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T} \text{cov}_t^\alpha = \alpha, \quad \text{for all } \alpha \in \mathcal{A}. \tag{1}$$

This ensures a coherence between forecasts and realized values, even if the distribution of the target changes over time.

2. *Distributional consistency.* Forecasts should also be ordered across quantile levels. That is, we want:

$$q_t^{\alpha_1} \le q_t^{\alpha_2} \le \cdots \le q_t^{\alpha_m}, \quad \text{for all } t = 1, 2, \ldots. \tag{2}$$

Without this ordering, the vector of forecasts would not correspond to a valid probability distribution, making it difficult for decision makers to interpret or trust.

There are many methods for producing quantile forecasts, including classical time series models such as ARIMA and exponential smoothing, as well as modern machine learning approaches such as random forests, and deep neural networks. However, these forecasts often fail to satisfy calibration. Our aim is to take any existing forecaster and transform its predictions *online* (in real time) so that the resulting forecasts satisfy both (1) and (2) for any sequence of outcomes. Henceforth, we refer to this joint goal as *calibration without crossings*. Furthermore, subject to calibration without crossings, we want the forecasts to remain *sharp*: all else equal, the forecasts should correspond to a probability distribution with low variance (the quantile predictions should not be too dispersed), so that they provide the least possible uncertainty about the outcome.

Our first objective, online calibration, has been studied extensively for the single ($m = 1$) quantile setting in the online conformal prediction literature, beginning with Gibbs and Candès (2021). Online conformal algorithms achieve distribution-free calibration (1) for a single level α. Of particular relevance to our paper is the quantile tracker (QT) algorithm from Angelopoulos et al. (2023). The idea behind this method is simple: to track the α-level quantile over time, we should increase our current quantile estimate if it is smaller than y_t (it "miscovers") and we should decrease our estimate if it is larger than or equal to y_t (it "covers"). The amount by which we increase or decrease these estimates is chosen to yield a long-run coverage of α, which is guaranteed whenever the target values are bounded in magnitude.

It is natural to try to use QT to solve the multi-level quantile calibration problem. However, simply applying this algorithm to multiple levels separately often results in quantile crossings, violating (2); in experiments on the COVID-19 Forecast Hub dataset from Cramer et al. (2022a), QT produced crossings at 87% of time steps on average (see Appendix A). To solve the problem of simultaneously calibrating multiple quantiles without producing crossings, we develop a procedure that we call the *multi-level quantile tracker (MultiQT)*, which combines a QT-style update for each level with an ordering step to ensure forecasted quantiles are distributionally consistent. As we later show, various naive ways of combining individual quantile calibration and ordering techniques do not achieve calibration, but our method provably does.

To derive the calibration guarantee for MultiQT, we first connect our goal of calibration without crossings to a more general problem of *constrained gradient equilibrium*. Many statistical objectives in online settings (including calibration) are special cases of a condition introduced by Angelopoulos et al. (2025) called *gradient equilibrium*, which says that the average of the loss function gradients evaluated at the chosen iterates converges to zero as the number of time steps goes to infinity. They show that to produce iterates which achieve gradient equilibrium, one can simply run online gradient descent, provided that the losses satisfy

certain weak conditions. However, it was heretofore not known whether gradient equilibrium can be achieved if the iterates must obey constraints, such as in our multi-level quantile forecasting setting, where our forecasts must lie in the set of ordered vectors. We provide an affirmative answer by showing that *lazy gradient descent*, which combines online gradient updates with a projection step to satisfy the iterate constraints, provably achieves gradient equilibrium as long as the loss function and constraint set jointly satisfy an additional condition we call *inward flow*.

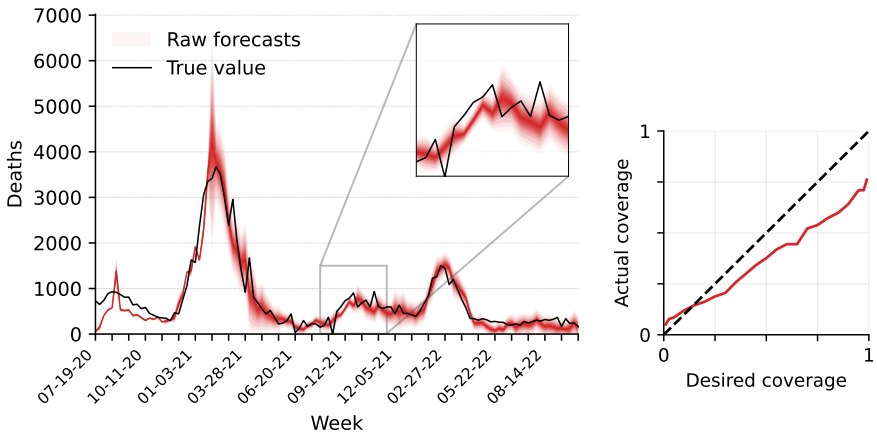
We show that the loss function and constraint set for the calibration without crossings problem satisfy inward flow. Thus, MultiQT, which can be written as lazy gradient descent on that loss function and constraint set, inherits a calibration guarantee from our more general analysis of constrained gradient equilibrium. Finally, we prove a no-regret guarantee for MultiQT with respect to the quantile loss. Due to the standard decomposition of the quantile loss into calibration and sharpness terms, this result can be informally interpreted as saying that MultiQT achieves calibration without paying a steep price in terms of sharpness.

1.1 A peek at results: calibrating COVID-19 forecasts

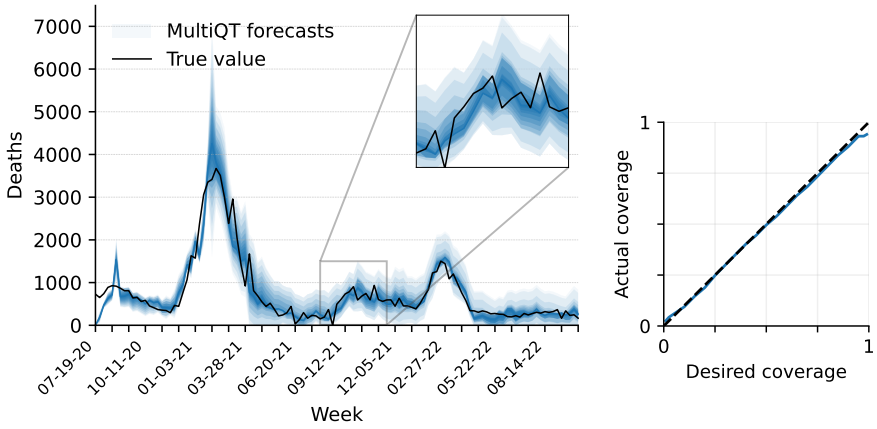
To illustrate the behavior of our method in practice, we begin with a brief case study. During the COVID-19 pandemic, forecasting teams submitted forecasts each week to the United States COVID-19 Forecast Hub of COVID-19 deaths in each state one, two, three, and four weeks into the future. In Figure 1a, we display one team's one-week-ahead forecasts for weekly COVID-19 deaths in California. We can see that these quantile forecasts are too narrow and biased downward; focusing in on the calibration plot shown in the right panel, we see that the forecasts fail to cover the true death count at the desired rate and convey more certainty than is appropriate. To remedy this, our proposed MultiQT method can be applied in real time to recalibrate such forecasts. Figure 1b shows the results of running MultiQT online, where at each time t, the method uses the performance of the forecasts up through time $t-1$ to correct the current forecast. We observe that MultiQT corrects the downward bias, particularly present in the upper quantiles, and the resulting forecasts achieve close to perfect calibration. By improving the coherence of the forecasts with eventual death counts, the use of a recalibration method such as MultiQT can improve the quality of public communication about the expected trajectory of pandemics and help inform timely public health decisions regarding allocation of scarce resources and hospital staffing (Cramer et al., 2022a). We will return to this COVID-19 forecasting application in Section 5.

1.2 Related work

Online calibration of a single quantile in the presence of distribution shift has been studied extensively in the context of online conformal prediction, beginning with Gibbs and Candès (2021). The central idea underlying many of these methods is to run online gradient descent on the quantile loss, applied to an iterate either in α-space (Gibbs and Candès, 2021) or in y-space (Angelopoulos et al., 2023). The latter has the advantage of leading to fewer infinite sets in general (as well as not requiring an expensive quantile computation at each time step). Later developments in this line of work include ways to adaptively set the learning rate (Zaffran et al., 2022; Gibbs and Candès, 2024), extensions to losses besides coverage (Feldman et al., 2022; Lekeufack et al., 2024), approaches tailored to multi-horizon forecasting (Yang et al., 2024; Wang and Hyndman, 2024) or that exploit error predictability (Hu et al., 2025), and methods that consider strongly adaptive regret (Bhatnagar et al., 2023; Hajihashemi and Shen, 2024). Gradient equilibrium, proposed in Angelopoulos



(a) Raw forecasts and their calibration.



(b) Forecasts and calibration after applying MultiQT.

Figure 1: One-week-ahead forecasts of weekly COVID-19 deaths in California from July 11, 2020 to October 22, 2022 by forecaster `RobertWalraven-ESG`, before (top) and after (bottom) applying MultiQT. Forecasts are made at 23 levels, roughly equally-spaced in between 0.01 and 0.99. To visualize these forecasts, we plot colored bands where the lightest opacity connects the 0.01 and 0.99 level forecasts, the next lightest connects the 0.025 and 0.975 level forecasts, and so on.

et al. (2025), generalizes the concept of online calibration to a setting with arbitrary losses, and in doing so, generalizes the analysis of online gradient descent that underlies work on online conformal prediction.

A complementary line of work on calibration uses Blackwell approachability and related ideas, resulting in procedures that are generally more complex than those based on gradient descent but also offer stronger (conditional) guarantees. The basic idea underpinning many papers on calibration (Foster, 1999; Foster and Hart, 2021) and defensive forecasting (Vovk et al., 2005; Perdomo and Recht, 2025) is that certain properties defined in terms of time-averages can be cast as special cases of Blackwell approachability. For example, the convex set in Blackwell approachability can be defined to encode zero calibration error. Gupta et al. (2022) builds on these ideas to develop algorithms for group-conditional quantile calibration, which are then applied to online conformal prediction in Bastani et al. (2022). This was later extended to the high-dimensional setting by Noarov et al. (2023). While this framework could in principle accommodate

multiple quantile calibration without crossings, it lacks a practical implementation for this problem setting, since it would require solving nontrivial inner optimization problems at each time step. In contrast, our approach is easily implementable and wraps around any existing forecaster, albeit targeting a weaker goal of unconditional calibration. Also related are Deshpande et al. (2023) and Marx et al. (2024), which use Blackwell approachability to calibrate probabilistic forecasts that specify a distribution over y_t.

Quantile prediction has a long history of study in statistics, dating back to the seminal work of Koenker and Bassett (1978). Though it is traditionally studied in the offline setting (with i.i.d. data), the problem of mitigating quantile crossing when jointly learning multiple quantiles has been present from the start (Bassett and Koenker, 1982). In the offline setting, solutions have been proposed in the form of post-processing (Chernozhukov et al., 2010; Fakoor et al., 2023), constrained optimization (Liu and Wu, 2009), or neural network learning architectures that enforce monotonicity of the output vector (Gasthaus et al., 2019; Park et al., 2022). In the online setting, Zhang et al. (2024) proposes a method that enforces monotonicity but achieves only a no-regret guarantee and not a calibration guarantee. Li and Rodríguez (2025) make use of ideas from Angelopoulos et al. (2023) to design a loss function for training a forecaster that targets coverage with no quantile crossings, but in practice their method still produces crossings (at roughly 10% of time steps in their experiments).

Finally, our work relates to a broader literature on forecast recalibration, which considers ways to improve the calibration of an existing forecaster (Brocklehurst et al., 1990) or an ensemble of forecasters (Hamill and Colucci, 1997; Raftery et al., 2005; Gneiting and Ranjan, 2013; van den Dool et al., 2017).

2 Methods

In this section, we present our online method for generating calibrated, distributionally consistent quantile forecasts given an arbitrary base forecaster. We do so by learning offsets that result in calibrated forecasts when added to the base forecasts. All omitted proofs in this and subsequent sections are deferred to Appendix B or C unless otherwise stated.

Notation. We use $b_t = (b_t^{\alpha_1}, b_t^{\alpha_2}, \ldots, b_t^{\alpha_m}) \in \mathbb{R}^m$ to denote the vector of base forecasts at time t, where b_t^α is the base forecast for level α. We use $\theta_t \in \mathbb{R}^m$ to denote the offset vector at time t (which we adjust online) and $q_t = b_t + \theta_t \in \mathbb{R}^m$ to denote the corresponding vector of recalibrated forecasts at time t. As with the base forecasts, we will often index elements of the offset and recalibrated forecast vectors by the quantile level, as in θ_t^α and q_t^α for a level α. We define $\mathrm{cov}_t^\alpha = \mathbb{1}\{y_t \le q_t^\alpha\}$ to be the coverage indicator for the α-level forecast. For a closed convex set $C \subseteq \mathbb{R}^d$, we use $\Pi_C(x) = \mathrm{argmin}_{z \in C} \|x - z\|_2^2$ to denote the projection of x onto C. We define $\mathcal{K} = \{x \in \mathbb{R}^d : x_1 \le x_2 \le \cdots \le x_d\}$ to be the d-dimensional isotonic cone, where the dimension d can be understood from context (in general, $d = m$). We refer to the projection $\Pi_\mathcal{K}$ onto $\mathcal{K}$ as *isotonic regression*.

Base forecasts. We assume the base forecasts are distributionally consistent:

$$b_t^{\alpha_1} \le b_t^{\alpha_2} \le \cdots \le b_t^{\alpha_m},$$

for all t. These base forecasts can be constructed in any way, e.g., $b_t^\alpha = f_t^\alpha(x_t)$ where f_t^α is some (possibly time-varying) predictor that optionally incorporates information from features x_t and is trained on past data (x_s, y_s), $s < t$. In problems with no base forecaster, we can set the base forecasts equal to zero (i.e., $b_t^\alpha = 0$

for all α and t). If instead there is a point forecaster that forecasts the mean or median μ_t at each time t, we can set the base forecasts to this point forecast (i.e., $b_t^\alpha = \mu_t$ for all α and t).

We begin by presenting some relevant background on the quantile tracker, which we then build on to present our proposed method, MultiQT.

2.1 Background: quantile tracker

Given a desired coverage level α, the *quantile tracker (QT)* method from Angelopoulos et al. (2023) works as follows. Given some initial offset $\theta_1^\alpha \in \mathbb{R}$ and learning rate $\eta > 0$, at each $t = 1, 2, \ldots$, we issue the adjusted forecast $q_t^\alpha = b_t^\alpha + \theta_t^\alpha$, observe y_t, and then update the offset according to:

$$\theta_{t+1}^\alpha = \theta_t^\alpha - \eta(\mathrm{cov}_t^\alpha - \alpha). \tag{3}$$

The update rule (3) is intuitive: we increase the offset by $\eta\alpha$ if we miscover, which makes it more likely we will cover at the following time step, and decrease the offset by $\eta(1-\alpha)$ if we cover.

The next result is from Proposition 1 of Angelopoulos et al. (2023). It shows that the QT is guaranteed to achieve long-run coverage, as long as the errors from the base forecaster are bounded.

Proposition 1 (Angelopoulos et al., 2023)**.** *Assume that $|y_t - b_t^\alpha| \le R$ for all t and some $R \ge 0$. Then, for all $T \ge 1$, the QT iterates (3) satisfy the coverage error bound*

$$\left| \frac{1}{T} \sum_{t=1}^{T} \mathrm{cov}_t^\alpha - \alpha \right| \le \frac{2|\theta_1^\alpha| + R + \eta}{\eta T}.$$

Consistent with this guarantee, the QT algorithm usually works well in practice. Unfortunately, applying the QT updates separately to *multiple* quantile levels often results in crossed quantiles, which is undesirable. A natural solution idea is to run QT separately for each level and then simply order the forecasts at each time step before revealing them to the user. Two ways of enforcing ordering are by sorting the given vector of quantile forecasts, or by applying isotonic regression. Perhaps surprisingly, neither one is able to achieve calibration in general, as the next result shows.

Proposition 2. *For a set $\mathcal{A}$ of m quantile levels, and for each $\alpha \in \mathcal{A}$, let q_t^α be obtained by the QT update rule (3). Given a map $G : \mathbb{R}^m \to \mathbb{R}^m$, let*

$$\hat{q}_t = G(q_t) \in \mathbb{R}^m$$

be the vector obtained by using G to post process the vector $q_t = (q^{\alpha_1}, q^{\alpha_2}, \ldots, q_t^{\alpha_m}) \in \mathbb{R}^m$ of QT forecasts at time t. Then, for both $G(v) = (v_{(1)}, \ldots, v_{(m)})$, which sorts the entries of its input, and $G(v) = \Pi_{\mathcal{K}}(v)$, which performs isotonic regression, there exists a set of quantile levels $\mathcal{A}$ and sequence of target values and base forecasts (y_t, b_t) with bounded errors (i.e., $|y_t - b_t^\alpha|$ is bounded for all α and t) such that for any learning rate $\eta > 0$, there is an $\alpha \in \mathcal{A}$ where $\lim_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T} \mathbb{1}\{y_t \le \hat{q}_t^\alpha\} \neq \alpha$ — that is, the α-level forecasts fail to achieve calibration.

The intuition for this result is simple: by Proposition 1, for each α, we know that the sequence q_t^α of QT iterates is guaranteed to achieve long-run coverage α. If we replace a nonvanishing fraction of the values in this sequence with some arbitrary value, then we should not expect the resulting sequence to still have coverage α. This is precisely what happens when crossings happen sufficiently often: whenever a crossing occurs, applying G maps $\hat{q}_t^\alpha$ to a value not equal to q_t^α (under sorting, it gets mapped to q_t^β for some $\beta \neq \alpha$, and under isotonic regression, it gets mapped to some local average). As a result, the long-run coverage of $\hat{q}_t^\alpha$ will differ from that of q_t^α. Based on this intuition, we construct a formal negative example in the appendix.

2.2 Multi-level quantile tracker

We now describe our method, called the *multi-level quantile tracker* (MultiQT), which adapts QT to the multiple quantile setting. This method is simple and, as we will later show, has compelling theoretical guarantees and strong empirical performance. At a high level, MultiQT maintains two vectors of offsets: one hidden and one played. The hidden offsets, denoted $\tilde{\theta}_t \in \mathbb{R}^m$, do not generally result in ordered forecasts when added to the base forecasts, but the played offsets, denoted $\theta_t \in \mathbb{R}^m$, do. MultiQT is described in Procedure 1.

Procedure 1. *Choose some initial value $\tilde{\theta}_1 \in \mathbb{R}^m$ and learning rate $\eta > 0$. For $t = 1, 2, \ldots$, repeat the following.*

1. *Compute the played offset $\theta_t = \Pi_{\mathcal{K}-b_t}(\tilde{\theta}_t)$.*
2. *Play the forecast $q_t = b_t + \theta_t$.*
3. *Observe y_t and update the hidden offset: for each $\alpha \in \mathcal{A}$,*

$$\tilde{\theta}^\alpha_{t+1} = \tilde{\theta}^\alpha_t - \eta(\mathrm{cov}^\alpha_t - \alpha). \tag{4}$$

Note that steps 1 and 2 can be combined into a single step:

$$q_t = \Pi_{\mathcal{K}}(b_t + \tilde{\theta}_t). \tag{5}$$

This is equivalent because $\Pi_C(x + b) = b + \Pi_{C-b}(x)$ for any closed convex set $C \subseteq \mathbb{R}^d$ and vectors $x, b \in \mathbb{R}^d$ (where $C - b = \{x - b : x \in C\}$). Writing the MultiQT forecast q_t in this way makes it clear that it belongs to $\mathcal{K}$ and is thus distributionally consistent. However, when running MultiQT in practice, it is convenient to implement each iteration as (5) followed by (4). When reasoning about its properties (calibration or regret), it is more convenient to use the form in Procedure 1. It is worth noting that each isotonic projection step $\Pi_{\mathcal{K}}$ can be computed efficiently in $O(m)$ time (where recall $m = |\mathcal{A}|$ is the number of quantile levels), using the pool adjacent violators algorithm (PAVA) (Ayer et al., 1955; Barlow et al., 1972).

We highlight that in (4) the hidden offset vector is updated based on the coverage induced by the played one: what appears in this update is $\mathrm{cov}^\alpha_t = \mathbb{1}\{y_t \le b^\alpha_t + \theta^\alpha_t\}$, rather than $\mathbb{1}\{y_t \le b^\alpha_t + \tilde{\theta}^\alpha_t\}$. More abstractly, the update takes a gradient step starting from the hidden offset but uses the gradient evaluated at the played offset. As we will see, this combination (known generally as lazy gradient descent) turns out to be crucial for achieving the desired calibration guarantee.

MultiQT with delayed feedback or lead time. Procedure 1 assumes that at each time t we are able to observe the outcome y_t before making our next forecast q_{t+1}. However, there are settings where this is not the case. We model such settings using a general framework of *delayed feedback*, in which the outcome associated with a forecast made at time t is revealed only after the forecast is made at time $t + D$ for some constant delay $D \ge 0$.

This framework naturally captures forecasting problems with a positive *lead time*, defined as the number of time steps between forecast issuance and outcome realization. For example, in weekly COVID-19 death forecasting, a four-week-ahead forecast has a lead time of four and corresponds to a feedback delay of $D = 3$. A lead time of one ($D = 0$) corresponds to the standard MultiQT setting, whereas $D \ge 1$ can be understood as a delayed feedback problem. The lead time is also referred to as the forecast horizon.

In the delayed feedback setting with delay $D \geq 0$, we can run a modification of MultiQT that is exactly like Procedure 1 except the hidden offset update in (4) is replaced with

$$\tilde{\theta}^{\alpha}_{t+1} = \tilde{\theta}^{\alpha}_{t} - \eta(\mathrm{cov}^{\alpha}_{t-D} - \alpha) \tag{6}$$

for $t > D$ and $\tilde{\theta}^{\alpha}_{t+1} = \tilde{\theta}^{\alpha}_{t}$ for $t \leq D$. In other words, at time t we update the hidden offset with the (delayed) feedback observed at time t, except for the first D time steps where no feedback is observed. Compared to the original MultiQT update (4), the difference is that the coverage indicator used in the above update corresponds to the forecast from time $t - D$, rather than time t.

3 Gradient equilibrium

To show that MultiQT solves the problem of calibration without crossings, we will first solve a more general problem we call *constrained gradient equilibrium* and then show that MultiQT is an instance of this general solution (Figure 2). Thinking about our problem at the more abstract gradient equilibrium level gives us a framework for cleanly proving the desired calibration guarantee.

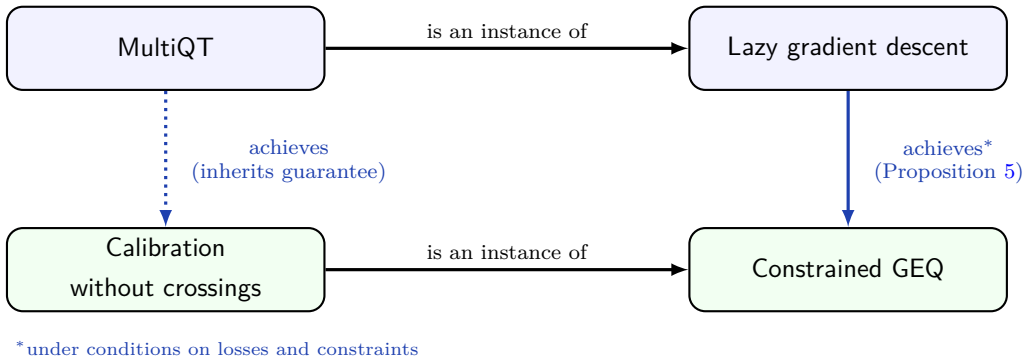


Figure 2: Illustration of the relationships between the MultiQT procedure, lazy gradient descent, calibration without crossings, and constrained gradient equilibrium.

We begin by recalling the definition of gradient equilibrium, from Angelopoulos et al. (2025).

Definition 1. *A sequence of iterates $\theta_t \in \mathbb{R}^d$, $t = 1, 2, \ldots$ is said to satisfy* gradient equilibrium *(GEQ) with respect to a sequence of real-valued loss functions ℓ_t, $t = 1, 2, \ldots$ if*

$$\lim_{T \to \infty} \frac{1}{T} \sum_{t=1}^{T} g_t(\theta_t) = \mathbf{0}, \tag{7}$$

where each $g_t(\theta)$ is a gradient (or subgradient) of ℓ_t at θ_t, assumed to be differentiable (or subdifferentiable) on its domain, and $\mathbf{0}$ is the d-dimensional zero vector.

There are many problems in online learning in which the iterates should be restricted to constraint sets (which may vary over time). This motivates the following definition.

Definition 2. *A sequence of iterates $\theta_t \in \mathbb{R}^d$, $t = 1, 2, \ldots$ is said to satisfy* constrained gradient equilibrium *(constrained GEQ) with respect to a sequence of loss functions ℓ_t, $t = 1, 2, \ldots$ and sets $C_t \subseteq \mathbb{R}^d$, $t = 1, 2, \ldots$ if gradient equilibrium (7) holds and, additionally, $\theta_t \in C_t$ for all $t = 1, 2, \ldots$.*

It is worth noting that, in optimization, it is common to reformulate a constraint set C via a characteristic function, denoted I_C (zero on C and ∞ otherwise) that is added to the loss ℓ_t and then treated as an unconstrained problem. However, constrained gradient equilibrium as we define it here is *not* the same as gradient equilibrium with respect to the modified loss sequence $\ell_t + I_{C_t}$. This is an important distinction that we will revisit shortly.

Calibration without crossings as constrained GEQ. For $\alpha \in [0,1]$, let $\rho_\alpha : \mathbb{R} \times \mathbb{R} \to \mathbb{R}$ be the α-level quantile loss, where

$$\rho_\alpha(\hat{y}, y) = \begin{cases} \alpha|y - \hat{y}| & \text{if } y \geq \hat{y} \\ (1-\alpha)|y - \hat{y}| & \text{otherwise.} \end{cases} \tag{8}$$

Given a set of levels $\mathcal{A}$ with $m = |\mathcal{A}|$, and a vector of forecasts $q \in \mathbb{R}^m$ at these levels, let $\rho_{\mathcal{A}} : \mathbb{R}^m \times \mathbb{R} \to \mathbb{R}$ be the aggregated quantile loss, where

$$\rho_{\mathcal{A}}(q, y) = \sum_{\alpha \in \mathcal{A}} \rho_\alpha(q^\alpha, y). \tag{9}$$

Now, for each t, define a loss function ℓ_t on $\theta_t \in \mathbb{R}^m$ that applies the aggregated quantile loss to $q_t = b_t + \theta_t$ and y_t, where $b_t \in \mathbb{R}^m$ is a vector of base forecasts:

$$\ell_t(\theta_t) = \rho_{\mathcal{A}}(b_t + \theta_t, y_t) = \sum_{\alpha \in \mathcal{A}} \rho_\alpha(b_t^\alpha + \theta_t^\alpha, y_t). \tag{10}$$

We will call (10) the *MultiQT loss*. A subgradient of the MultiQT loss at θ_t is

$$g_t(\theta_t) = \begin{bmatrix} \mathrm{cov}_t^{\alpha_1} - \alpha_1 \\ \mathrm{cov}_t^{\alpha_2} - \alpha_2 \\ \vdots \\ \mathrm{cov}_t^{\alpha_m} - \alpha_m \end{bmatrix} \tag{11}$$

where we recall that $\mathrm{cov}_t^\alpha = \mathbb{1}\{y_t \leq q_t^\alpha\} = \mathbb{1}\{y_t \leq b_t^\alpha + \theta_t^\alpha\}$. To streamline presentation, we will often refer to (11) as the "gradient" of the MultiQT loss. We now observe the following equivalence:

$$\lim_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T} g_t(\theta_t) = \mathbf{0} \quad \Longleftrightarrow \quad \lim_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T} \mathrm{cov}_t^\alpha = \alpha, \text{ for all } \alpha \in \mathcal{A}.$$

In other words, for a sequence of quantile forecasts q_t, $t = 1, 2, \ldots$ to be calibrated, it suffices to show that the offsets θ_t, $t = 1, 2, \ldots$ satisfy gradient equilibrium with respect to the MultiQT loss defined in (10).

Furthermore, recall that our goal is to derive offsets that, once added to the base forecasts, yield forecasts that are not only calibrated but are also distributionally consistent. Setting the constraint set at time t as

$$C_t = \mathcal{K} - b_t, \tag{12}$$

which is the isotonic cone shifted by the base forecast b_t, ensures that the resulting forecast q_t does not have crossed quantiles. Thus, calibration without crossings is an instance of constrained gradient equilibrium, for the losses and constraints defined above.

3.1 Background: gradient descent

Before solving the constrained gradient equilibrium problem, we first consider the (unconstrained) gradient equilibrium problem. It turns out that we do not need to devise new algorithms in order to produce iterates that satisfy gradient equilibrium. Online gradient descent, a standard algorithm in online learning, can also be used to solve the gradient equilibrium problem.

Gradient descent achieves GEQ. Given some initial point $\theta_1 \in \mathbb{R}^d$ and learning rate $\eta > 0$, recall that *online gradient descent*, which we will often simply call gradient descent (GD), obtains iterates via

$$\theta_{t+1} = \theta_t - \eta g_t(\theta_t), \tag{13}$$

for $t = 1, 2, \ldots$, where $g_t(\theta_t)$ is a (sub)gradient of the loss at θ_t. As explained in Angelopoulos et al. (2025), we can rearrange (13) to get $g_t(\theta_t) = (\theta_t - \theta_{t+1})/\eta$ and then average over t to yield

$$\frac{1}{T}\sum_{t=1}^{T} g_t(\theta_t) = \frac{\theta_1 - \theta_{T+1}}{\eta T}. \tag{14}$$

Since θ_1 is chosen by us, it is bounded. Thus if we can bound θ_{T+1}, this would imply a bound on the average gradient. Angelopoulos et al. (2025) show that a sufficient condition for θ_{T+1} to be bounded or sublinear in T is for the loss functions to satisfy two conditions. The first is Lipschitzness; this is a standard condition, and we recall that a loss ℓ is said to be *L-Lipschitz* if for all θ, all of its subgradients $g(\theta)$ satisfy $\|g(\theta)\|_2 \leq L$. The second is a new condition that they call restorativity, which we describe below.

Definition 3. *A loss ℓ is said to be* (h, ϕ)-restorative*, for a constant $h \geq 0$ and nonnegative function ϕ, if it has a subgradient $g(\theta)$ at each θ which satisfies*

$$\langle \theta, g(\theta) \rangle \geq \phi(\theta), \quad \text{whenever } \|\theta\|_2 > h, \tag{15}$$

where $\langle u, v \rangle = u^\mathsf{T} v$.

Intuitively, restorativity (15) tells us that whenever the iterates get too far from the origin, the negative gradient will push the iterate back towards the origin. This can be seen most easily in the one-dimensional setting where $\theta \in \mathbb{R}$ and $\phi(\theta) = 0$: in this case, restorativity says that $\text{sign}(\theta) = \text{sign}(g(\theta))$ whenever $|\theta| \geq h$. In other words, if θ is large in magnitude, then the negative gradient will be anti-aligned with it, so following the negative gradient will decrease the magnitude of θ.

This intuition is formalized in the following result, which appears as Proposition 5 in Angelopoulos et al. (2025). It says that gradient descent produces iterates that grow slowly when the losses are restorative.

Proposition 3 (Angelopoulos et al., 2025)**.** *Assume that for each t, the loss function ℓ_t is L-Lipschitz and $(h_t, 0)$-restorative. Then, for all $T \geq 1$, the gradient descent iterates produced according to (13) satisfy*

$$\|\theta_{T+1}\|_2 \leq \sqrt{\|\theta_1\|_2^2 + \eta^2 L^2 T + 2\eta L \sum_{t=1}^{T} h_t}.$$

If h_t is nondecreasing, then this implies

$$\left\| \frac{1}{T}\sum_{t=1}^{T} g_t(\theta_t) \right\|_2 \leq \frac{2\|\theta_1\|_2}{\eta T} + \sqrt{\frac{L^2}{T} + \frac{2Lh_T}{\eta T}},$$

which goes to zero as $T \to \infty$ as long as h_T is sublinear in T.

When all loss functions ℓ_t are $(h, 0)$-restorative, note that the rate at which gradient equilibrium is obtained in Proposition 3 is $1/\sqrt{T}$. Angelopoulos et al. (2025) show that gradient equilibrium rates of order $1/T$ are possible under stronger conditions, including when ϕ is lower bounded by a positive constant (rather than zero, as assumed in Proposition 3). They also show that the one-dimensional case is special: when $d = 1$, L-Lipschitzness and $(h, 0)$-restorativity are sufficient for achieving the fast $1/T$ rate.

By the calculations at the start of this section (8)–(11) specialized to the singleton $\mathcal{A} = \{\alpha\}$, we can see that the QT update (3) is simply online gradient descent with respect to the loss $\ell_t(\theta) = \rho_\alpha(b_t^\alpha + \theta, y_t)$. This loss is Lipschitz with $L = 1$ and, under bounded errors $|y_t - b_t^\alpha| \leq R$, it can be shown to be $(R, 0)$-restorative. Hence, the calibration guarantee for QT can be derived as a special case of Proposition 3. Indeed, the exact result in Proposition 1 (which shows calibration is achieved at the rate $1/T$) can be derived as a special case of the one-dimensional refinement of Proposition 3. We refer to Corollary 1 of Angelopoulos et al. (2025).

Projected gradient descent does not achieve constrained GEQ. Perhaps the most common way to enforce iterate constraints is via projection. Now that we have seen gradient descent achieves gradient equilibrium (under some mild conditions), a natural first guess for achieving constrained gradient equilibrium would be to run projected gradient descent. Given closed convex constraint sets C_t, $t = 1, 2, \ldots$, an initial $\theta_1 \in C_1$, and learning rate $\eta > 0$, *projected gradient descent* obtains iterates via the update rule:

$$\theta_{t+1} = \Pi_{C_{t+1}}(\theta_t - \eta g_t(\theta_t)), \tag{16}$$

for $t = 1, 2, \ldots$. Unfortunately, projected gradient descent does not guarantee constrained gradient equilibrium in general, and, in fact, provably fails to achieve our goal of calibration without crossings.

To see why, first observe that we can view projected gradient descent as ordinary gradient descent on the modified loss sequence $\tilde{\ell}_t = \ell_t + I_{C_t}$, where

$$I_{C_t}(\theta) = \begin{cases} 0 & \text{if } \theta \in C_t \\ \infty & \text{otherwise} \end{cases}$$

is the characteristic function of C_t. Subgradients of $\tilde{\ell}_t$ at θ are of the form $\tilde{g}_t = g_t(\theta) + v_t(\theta)$, where $v_t(\theta)$ is a subgradient of I_{C_t} at θ, i.e., an element of the normal cone of C_t at θ.

Next, as noted in Appendix B of Angelopoulos et al. (2025), due to the gradient equilibrium guarantee of gradient descent (Proposition 3), the projected gradient descent iterates will achieve gradient equilibrium with respect to this modified sequence, meaning $\lim_{T\to\infty} \frac{1}{T} \sum_{t=1}^T \tilde{g}_t(\theta_t) = 0$. This implies

$$\lim_{T\to\infty} \frac{1}{T} \sum_{t=1}^T g_t(\theta_t) = -\lim_{T\to\infty} \frac{1}{T} \sum_{t=1}^T v_t(\theta_t).$$

Since the right-hand side is not zero in general, projected gradient descent is not guaranteed to achieve constrained gradient equilibrium. The next proposition goes further and shows that projected gradient descent provably fails to solve our calibration without crossings problem.

Proposition 4. *There exists a set of levels $\mathcal{A}$ and sequence of target values and base forecasts (y_t, b_t) with bounded errors (i.e., $|y_t - b_t^\alpha|$ is bounded for all α and t) such that for any learning rate $\eta > 0$, projected gradient descent (16), with the gradient g_t as defined in (11) and constraint set C_t as defined in (12), fails to achieve calibration: $\lim_{T\to\infty} \frac{1}{T} \sum_{t=1}^T \mathbb{1}\{y_t \leq b_t^\alpha + \theta_t^\alpha\} \neq \alpha$ for some $\alpha \in \mathcal{A}$.*

3.2 Lazy gradient descent

As we saw above, incorporating constraints into gradient descent in the standard way (via direct projection at each step), fails to achieve constrained gradient equilibrium. Another way to incorporate constraints into online gradient descent is to use what are known as lazy updates (Shalev-Shwartz, 2012; Hazan, 2019). Presented with the same constraint sets as in projected gradient descent, we implement lazy updates by maintaining two sequences: a hidden sequence $\tilde{\theta}_t$ and a played sequence θ_t. More precisely, *lazy online gradient descent* (lazy GD) begins with an initial point $\tilde{\theta}_1 \in C_1$ and learning rate $\eta > 0$, and obtains iterates via a two-step procedure:

$$\theta_t = \Pi_{C_t}(\tilde{\theta}_t), \tag{17}$$

$$\tilde{\theta}_{t+1} = \tilde{\theta}_t - \eta g_t(\theta_t). \tag{18}$$

"Lazy" refers to how (18) takes the gradient step starting from the hidden iterate $\tilde{\theta}_t$ rather than the played iterate θ_t. If we instead took the gradient step starting from θ_t in (18), then this would be equivalent to projected gradient descent (16).

The utility of the lazy updates can be seen immediately; by rearranging (18) and averaging over t, just as in unconstrained gradient descent, we get

$$\frac{1}{T}\sum_{t=1}^{T} g_t(\theta_t) = \frac{\tilde{\theta}_1 - \tilde{\theta}_{T+1}}{\eta T}. \tag{19}$$

This calculation leverages the fact that in lazy gradient descent we have effectively decoupled the updates of the hidden iterates from projection, and thus to track the average gradient, we can track total movement in the hidden sequence. The only difference from the previous result (14) for ordinary gradient descent is that the hidden iterates appear on the right-hand side in (19), rather than the played iterates.

Thus, to bound the average gradients of lazy gradient descent, we want to bound the hidden iterates. Recall that to control the growth of the played iterates in standard gradient descent, we controlled the inner product between θ_t and $g_t(\theta_t)$ via restorativity. To control the growth of the hidden iterates of lazy gradient descent, the relevant inner product is now between $\tilde{\theta}_t$ and $g_t(\theta_t)$. Note carefully that we seek to measure the alignment of an iterate $\tilde{\theta}_t$ with the gradient at a *different* point: the result θ_t of projection onto the constraint set C_t. Restorativity of the loss alone is not sufficient for this purpose. We need to introduce an additional condition that controls the joint behavior of the loss and the constraint set.

Definition 4. *For a loss ℓ and set C, the pair (ℓ, C) is said to satisfy* inward flow *if there is a subgradient $g(\theta)$ at each θ that satisfies*

$$-g(\theta) \in T_C(\theta), \quad \text{for } \theta \in \mathrm{bd}(C), \tag{20}$$

where $T_C(x)$ denotes the tangent cone of C at x, defined as

$$T_C(x) = \mathrm{cl}\big\{y : \text{there exists } \delta > 0 \text{ such that } x + \varepsilon y \in C \text{ for all } \varepsilon \in (0, \delta]\big\}.$$

In the above, $\mathrm{bd}(S)$ denotes the boundary of a set S, and $\mathrm{cl}(S)$ denotes the closure of S.

Informally, inward flow (20) says that if we start at any θ on the boundary of the constraint set and take an arbitrarily small step in the direction of the negative gradient, then this will keep us within the constraint set. In other words, following the direction of steepest descent will lead us inward, "flowing" further into the constraint set. Figure 3 provides a visualization.

Combining both restorativity and inward flow, we are able to establish the following result.

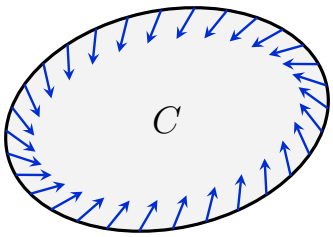


Figure 3: Visualization of inward flow, with the arrows representing the negative gradient $-g$ evaluated at different points on the boundary of the constraint set C.

Proposition 5. *Assume that for each t, the loss function ℓ_t is L-Lipschitz and $(h_t, 0)$-restorative, the set C_t is closed and convex, and the pair (ℓ_t, C_t) satisfies inward flow. Then, for all $T \geq 1$, the lazy gradient descent iterates produced by (17) and (18) satisfy*

$$\|\tilde{\theta}_{T+1}\|_2 \leq \sqrt{\|\tilde{\theta}_1\|_2^2 + \eta^2 L^2 T + 2\eta L \sum_{t=1}^{T} h_t}.$$

If h_t is nondecreasing, then this implies

$$\left\| \frac{1}{T} \sum_{t=1}^{T} g_t(\theta_t) \right\|_2 \leq \frac{2\|\tilde{\theta}_1\|_2}{\eta T} + \sqrt{\frac{L^2}{T} + \frac{2Lh_T}{\eta T}},$$

which goes to zero as $T \to \infty$ as long as h_T is sublinear in T.

We remark that Proposition 3 can be recovered as a special case of the above result: gradient descent is an instance of lazy gradient descent where the constraint set at every time t is $C_t = \mathbb{R}^d$, thus the projection is the identity, and $\theta_t = \tilde{\theta}_t$. Since $\mathbb{R}^d$ has no boundary, inward flow is trivially satisfied, and in this way Proposition 5 exactly reduces to Proposition 3.

The above result can also be extended to the delayed feedback setting. With delay $D \geq 0$, we can generalize lazy gradient descent and update the hidden iterates according to:

$$\tilde{\theta}_{t+1} = \tilde{\theta}_t - \eta g_{t-D}(\theta_{t-D}), \tag{21}$$

where we set $g_t(\theta_t) = 0$ for $t \leq 0$. We maintain the projection step (17) for obtaining the played iterates.

Proposition 6. *Under the conditions of Proposition 5, in the setting with feedback delay $D \geq 0$, the lazy gradient descent iterates produced by (17) and (21) satisfy*

$$\|\tilde{\theta}_{T+D+1}\|_2 \leq \sqrt{\|\tilde{\theta}_1\|_2^2 + \eta^2 L^2 (2D+1) T + 2\eta L \sum_{t=1}^{T} h_t}.$$

If h_t is nondecreasing, then this implies

$$\left\| \frac{1}{T} \sum_{t=1}^{T} g_t(\theta_t) \right\|_2 \leq \frac{2\|\tilde{\theta}_1\|_2}{\eta T} + \sqrt{\frac{L^2(2D+1)}{T} + \frac{2Lh_T}{\eta T}},$$

which goes to zero as $T \to \infty$ as long as h_T is sublinear in T.

Summary of positive and negative results on constrained GEQ. In Table 1 we review the results we have established thus far on constrained gradient equilibrium from Propositions 2, 4, and 5. Post hoc projection is the method from Proposition 2, with $G = \Pi_{\mathcal{K}}$; recall in that proposition we showed it fails to attain calibration without crossings, and hence it fails to achieve constrained gradient equilibrium in general. Projected gradient descent similarly fails according to Proposition 4, whereas lazy gradient descent achieves constrained gradient equilibrium (under restorativity and inward flow) by Proposition 5.

Table 1: Summary of results on online methods that incorporate gradient updates and projection onto constraints. Note that the post hoc projection and projected gradient descent methods are rewritten here using a hidden sequence to make the differences between methods more salient.

	Post hoc projection	**Projected GD**	**Lazy GD**
Projection	$\theta_t = \Pi_{C_t}(\tilde{\theta}_t)$		
Hidden update	$\tilde{\theta}_{t+1} = \tilde{\theta}_t - \eta_t g_t(\tilde{\theta}_t)$	$\tilde{\theta}_{t+1} = \theta_t - \eta_t g_t(\theta_t)$	$\tilde{\theta}_{t+1} = \tilde{\theta}_t - \eta_t g_t(\theta_t)$
Enforces constraint?	✓	✓	✓
Achieves GEQ?	×	×	✓*

*under restorativity and inward flow

Why does lazy gradient descent succeed in achieving constrained gradient equilibrium, while the other two methods fail? Here we give some intuition. First, *post hoc projection discards current information*, as feedback from the played iterate is never incorporated into subsequent updates. Specifically, the observed gradient $g_t(\theta_t)$ is not used to inform future updates, and this turns out to be problematic when the goal is to drive the average of such gradients to zero. Second, *projected gradient descent discards past information*, since the hidden update does not depend on the hidden iterate $\tilde{\theta}_t$. The hidden iterate encodes accumulated knowledge about the gradients of interest over the sequence thus far (recall (19)), and "forgetting" this information again turns out to be problematic. Observe that *lazy gradient descent combines both sources of information:* it preserves past knowledge by starting its update from $\tilde{\theta}_t$, while at the same time incorporating current feedback via the gradient evaluated at θ_t. This blend of retaining history and responding to present information is what distinguishes lazy gradient descent and enables it to achieve constrained gradient equilibrium.

4 MultiQT theory

Having introduced the framework of constrained gradient equilibrium in the last section, we are now ready to present the theoretical guarantees for MultiQT. Recall the proof roadmap illustrated in Figure 2, and note that we have already shown (i) calibration without crossings is an instance of constrained gradient equilibrium, and (ii) lazy gradient descent solves constrained gradient equilibrium problems that satisfy Lipschitz, restorativity, and inward flow conditions. What remains to be shown are (iii) MultiQT is the appropriate instantiation of lazy gradient descent for the problem of calibration without crossings, and (iv) the calibration without crossings problem satisfies the needed conditions (Lipschitz, restorativity, and inward flow). We address (iii) in the next paragraph, and (iv) in the following subsection. After this, a calibration

guarantee for MultiQT will follow directly from the gradient equilibrium theory developed in the previous section.

MultiQT is lazy gradient descent. This follows directly from the calculations already given at the start of Section 3. Referring back to Procedure 1, we can see that the hidden update (4) is equivalent to (18) with respect to the MultiQT gradient in (11), and the played update is equivalent to projection onto C_t as in (12) (indexed slightly differently because Procedure 1 is clearer in the context of forecasting).

4.1 Calibration guarantee

It remains to show that the MultiQT losses and constraints satisfy the Lipschitz, restorativity, and inward flow conditions. Lipschitzness is straightforward to show: examining the gradient in (11), we see that

$$\|g_t(\theta)\|_2 = \sqrt{\sum_{\alpha \in \mathcal{A}} (\mathrm{cov}_t^\alpha - \alpha)^2} \leq \sqrt{\sum_{\alpha \in \mathcal{A}} 1} = \sqrt{m},$$

so each loss is $\sqrt{m}$-Lipschitz.

The second condition, restorativity, is satisfied by the MultiQT losses as long as the errors between the base forecast and target values are bounded, as the following lemma establishes.

Lemma 1. *Assume that $|y_t - b_t^\alpha| \leq R$ for all α. Let $d_{\mathcal{A}} = \min_{\alpha \in \mathcal{A}} \min\{\alpha, 1-\alpha\}$ be the minimum distance between any level in $\mathcal{A}$ and the boundary of $[0,1]$. Then the MultiQT loss defined in (10) is $(h, 0)$-restorative for any $h \geq Rm^{3/2}/d_{\mathcal{A}}$, where recall $m = |\mathcal{A}|$.*

We next verify the third condition, inward flow, for the MultiQT loss and (shifted) isotonic cone.

Lemma 2. *The MultiQT loss defined in (10), with gradient in (11), and the constraint C_t defined in (12), together satisfy inward flow: $-g_t(\theta) \in T_{C_t}(\theta)$ for all θ on the boundary of C_t.*

Before moving on, we reflect on the above two lemmas. While restorativity of the MultiQT loss is to be expected based on results from the single quantile case (Angelopoulos et al., 2025), the fact that the MultiQT loss satisfies inward flow over the isotonic cone is perhaps more surprising. Inward flow is highly nontrivial, and requires the gradients of the loss to "cooperate" with the geometry of the constraint set. This can fail to hold even in seemingly natural optimization problems, which is a point we return to in Section 6.

We now state the main result for MultiQT, which follows directly from the previous results.

Theorem 1. *Assume that $|y_t - b_t^\alpha| \leq R$ for all α, t. Then, for all $T \geq 1$, the MultiQT iterates from Procedure 1 satisfy the ℓ_2 coverage error bound*

$$\sqrt{\sum_{\alpha \in \mathcal{A}} \left(\frac{1}{T} \sum_{t=1}^T \mathrm{cov}_t^\alpha - \alpha \right)^2} \leq \frac{2\|\tilde{\theta}_1\|_2}{\eta T} + \sqrt{\frac{m}{T} + \frac{2Rm^2}{\eta d_{\mathcal{A}} T}},$$

where recall $m = |\mathcal{A}|$, and $d_{\mathcal{A}}$ is as in Lemma 1. Since $\|x\|_\infty \leq \|x\|_2$ for any vector $x \in \mathbb{R}^m$, the right-hand side above is also an upper bound for $\left| \frac{1}{T} \sum_{t=1}^T \mathrm{cov}_t^\alpha - \alpha \right|$, for each $\alpha \in \mathcal{A}$.

Proof. We apply Proposition 5 to the current problem setting. Its conditions are verified by Lemmas 1 and 2, with $L = \sqrt{m}$ and $h = Rm^{3/2}/d_{\mathcal{A}}$. □

Theorem 1 tells us that MultiQT is guaranteed to achieve calibration, as described in (1). Moreover, the projection step ensures that the forecasts satisfy the distributional consistency property from (2). Thus, we have shown that the MultiQT method is guaranteed to satisfy our initial desiderata. This is true even in the delayed feedback setting, as the next generalization shows.

Theorem 2. *Under the conditions of Theorem 1, in the setting with feedback delay $D \geq 0$, the MultiQT iterates from Procedure 1 with the modification in (6) satisfy*

$$\sqrt{\sum_{\alpha \in \mathcal{A}} \left(\frac{1}{T} \sum_{t=1}^{T} \mathrm{cov}_t^{\alpha} - \alpha \right)^2} \leq \frac{2\|\tilde{\theta}_1\|_2}{\eta T} + \sqrt{\frac{m(2D+1)}{T} + \frac{2Rm^2}{\eta d_{\mathcal{A}} T}}.$$

Proof. We apply Proposition 6 to the current problem setting. Its conditions are again verified by Lemmas 1 and 2. □

We can see from the above theorem that the bound worsens with increasing delay, which is unsurprising.

4.2 Regret guarantee

In this subsection, we provide a regret guarantee for MultiQT with respect to the MultiQT loss (the aggregated quantile loss). Beyond being the loss we take gradient updates with respect to, its relevance can be motivated by the fact that it admits a decomposition into terms that can be interpreted as emphasizing calibration and sharpness. To be precise, suppose that the set $\mathcal{A}$ of quantile levels is symmetric around 1/2, and can therefore be written as

$$\mathcal{A} = \bigcup_{\beta \in \mathcal{B}} \{\beta/2,\, 1 - \beta/2\},$$

for some set $\mathcal{B} \subset [0, 1/2]$. Then it can be shown (Bracher et al., 2021) that the aggregated quantile loss $\rho_{\mathcal{A}}$ in (9) has the following alternative representation:

$$\rho_{\mathcal{A}}(q, y) = \sum_{\beta \in \mathcal{B}} \Bigg[\underbrace{\mathrm{dist}(y, [q_{\beta/2}, q_{1-\beta/2}])}_{\text{“calibration”}} + \underbrace{\frac{\beta}{2}(q_{1-\beta/2} - q_{\beta/2})}_{\text{“sharpness”}} \Bigg], \tag{22}$$

where $\mathrm{dist}(y, [a, b])$ is zero if y lies inside $[a, b]$, and otherwise equals the distance to the closest endpoint. The expression on the right-hand side corresponds to the *weighted interval score* of a collection of equi-tailed prediction intervals $[q_{\beta/2}, q_{1-\beta/2}]$, $\beta \in \mathcal{B}$. In each summand, the first term—which measures the distance of the target y to the interval $[q_{\beta/2}, q_{1-\beta/2}]$—can be interpreted as a calibration penalty, whereas the second term—which measures the length of the interval—can be interpreted as a sharpness penalty.

Given that we have already shown via gradient equilibrium theory that the MultiQT method achieves calibration (which as we have defined it, means long-run coverage per quantile level), the regret theory below can be interpreted in light of the decomposition (22) as saying that MultiQT-adjusted forecasts also encourage sharpness, i.e., they give rise to prediction intervals that are as concentrated and informative as possible. Moreover, the quantile loss is a proper scoring rule for quantile forecasts (Gneiting and Raftery, 2007; Gneiting et al., 2023), commonly used in the applied forecasting community, and regret results with respect to quantile loss may therefore be meaningful in their own right.

We now turn to our regret guarantee for MultiQT. We will study

$$\frac{1}{T} \sum_{t=1}^{T} \ell_t(\theta_t) - \inf_{\theta \in \mathbb{R}^m} \frac{1}{T} \sum_{t=1}^{T} \ell_t(\theta),$$

called the (average) *regret* of MultiQT iterates θ_t with respect to the MultiQT losses ℓ_t defined in (10). The analysis of regret in this setting is somewhat nonstandard, because of the time-varying constraints C_t, $t = 1, 2, \ldots$. In the appendix we derive a more general regret bound for problems with constraints satisfying inward flow, of which the following is a consequence.

Theorem 3. *Assume that $|y_t - b_t^\alpha| \leq R$ for all α and t and define $\ell_t(\theta) = \rho_{\mathcal{A}}(b_t + \theta, y_t)$. Then, for all $T \geq 1$, the MultiQT iterates from Procedure 1 satisfy the regret bound*

$$\frac{1}{T}\sum_{t=1}^{T} \ell_t(\theta_t) - \inf_{\theta \in \mathbb{R}^m} \frac{1}{T}\sum_{t=1}^{T} \ell_t(\theta) \leq \frac{\|\tilde{\theta}_1\|_2^2}{\eta T} + \frac{R^2 m}{\eta T} + \frac{\eta L^2}{2}. \tag{23}$$

When $\tilde{\theta}_1 = 0$, the learning rate that minimizes the right-hand side in (23) is $\eta = R\sqrt{2m}/(L\sqrt{T})$; plugging this in gives the bound

$$\frac{1}{T}\sum_{t=1}^{T} \ell_t(\theta_t) - \inf_{\theta \in \mathbb{R}^m} \frac{1}{T}\sum_{t=1}^{T} \ell_t(\theta) \leq \frac{RL\sqrt{2m}}{\sqrt{T}}. \tag{24}$$

This is a no-regret result: the right-hand side converges to zero as $T \to \infty$, at the rate $1/\sqrt{T}$. Note that we can take as the comparator the vector of all zeros ($\theta = \mathbf{0}$), hence (24), or more generally (23), also bounds the excess average quantile loss suffered by MultiQT compared to *no adjustment*, i.e., compared to that incurred by the original base forecasts.

Furthermore, the above result bounds the regret of MultiQT iterates compared to the vector θ^* of empirical quantiles in hindsight. More precisely, each entry $\theta^{*,\alpha}$ is an α-level empirical quantile of $y_t - b_t^\alpha$, $t = 1, \ldots, T$. In fact, θ^* is the optimal comparator — that is, it minimizes the average loss through time T.

The next theorem generalizes the previous one to the setting of delayed feedback.

Theorem 4. *Under the conditions of Theorem 3, in the setting with feedback delay $D \geq 0$, the MultiQT iterates from Procedure 1 with the modification in (6) satisfy*

$$\frac{1}{T}\sum_{t=1}^{T} \ell_t(\theta_t) - \inf_{\theta \in \mathbb{R}^m} \frac{1}{T}\sum_{t=1}^{T} \ell_t(\theta) \leq \frac{\|\tilde{\theta}_1\|_2^2}{\eta T} + \frac{R^2 m}{\eta T} + \frac{\eta(2D+1)L^2}{2}.$$

4.3 Calibration-regret tradeoff?

When the learning rate η is chosen appropriately, the bounds in Theorem 1 and Theorem 3 guarantee that the ℓ_2 calibration error and regret, respectively, vanish at the rate $1/\sqrt{T}$. However, the choice of learning rate is pulled in opposite directions by these results: the theoretical bounds tell us that calibration improves with larger η, whereas regret improves with smaller η.

To balance the bounds provided in these theorems, we can identify the dominant terms: $O(1/\sqrt{\eta T})$ in Theorem 1, versus $O(\eta)$ in Theorem 3. Equating these two leads to the choice of learning rate $\eta = O(T^{-1/3})$, which then yields a $O(T^{-1/3})$ bound on both calibration error and regret.

This leads to an interesting question: is this tradeoff fundamental? That is, must any method necessarily trade off calibration and regret (as we have defined them here)? And if so, is $O(T^{-1/3})$ the optimal common rate at which they both can be controlled?

More refined results on MultiQT calibration error, which we will describe later in the discussion, suggest that the answer to the latter question may be no in general, and most certainly no in a more specialized case. When the base forecasts are point forecasts (i.e., $b_t^\alpha = \mu_t$ for all α and t), we can obtain faster rates for the calibration error of MultiQT, with dominant term $O(1/(\eta T))$; balancing this with the regret bound

leads to a choice of learning rate $\eta = O(T^{-1/2})$, which provides a $O(T^{-1/2})$ bound on calibration error and regret. Whether this can be extended beyond point forecasts is an open question, as is the general tradeoff between calibration and regret (or gradient equilibrium error and regret, even more generally).

5 Experiments

We apply the MultiQT method to two real forecasting datasets relating to COVID-19 deaths and renewable energy production.[1] To evaluate calibration, we will investigate plots of the actual (empirical) coverage

$$\widehat{\text{cov}}^{\alpha} = \frac{1}{T}\sum_{t=1}^{T} \text{cov}_t^{\alpha}$$

versus the desired (nominal) coverage α, over the given set $\mathcal{A}$ of quantile levels. We will also examine the ℓ_1 calibration error, normalized by the number of levels $m = |\mathcal{A}|$, defined as

$$\text{Calibration error} = \frac{1}{m}\sum_{\alpha\in\mathcal{A}} |\widehat{\text{cov}}^{\alpha} - \alpha|.$$

In the applied forecasting literature, it is common to measure miscalibration by first computing the probability integral transform (PIT) values associated with the forecast distributions and target values over time and then reporting the entropy of these PIT values (Gneiting et al., 2007; Rumack et al., 2022). As this metric is not specific to quantile forecasts and requires a conversion to the cumulative density function, we primarily study calibration error as defined in the above display, but we provide results using PIT entropy in Appendix E.1 for completeness.

We additionally study the quantile loss, normalized by the number of levels, defined as

$$\text{Quantile loss} = \frac{1}{Tm}\sum_{t=1}^{T} \rho_{\mathcal{A}}(q_t, y_t).$$

As explained in Section 4.2, this loss function is proper, emphasizes sharpness, and is commonly used in the forecasting community (where it is often written in an equivalent form, called the weighted interval score).

Lastly, to set the learning rate η in MultiQT, we modify a heuristic proposed in Angelopoulos et al. (2023) for the learning rate in QT. They set the learning rate adaptively: the learning rate at time t is set to be 0.1 times the largest absolute error $|y_s - b_s^{\alpha}|$ seen in the last 50 time steps $s = t-50, \ldots, t-1$. We replace this max of recent errors with a 90% quantile to avoid setting excessively large learning rates after encountering a single large error. Specifically, we set the learning rate at time t as

$$\eta_t = \max\left\{0.1 \cdot \text{Quantile}_{0.9}\left(\bigcup_{\alpha\in\mathcal{A}} \{|y_s - b_s^{\alpha}|\}_{s=t-50}^{t-1}\right), \epsilon\right\},$$

for $\epsilon = 0.1$. The lower limit of ϵ ensures that the learning rate is positive even if the residuals are zero.

5.1 COVID-19 death forecasting

During the COVID-19 pandemic, forecasts of the pandemic's trajectory were used to help guide short-term decisions relating to policy and resource allocation, as well as for general public communication. The United States COVID-19 Forecast Hub is a repository of forecasts made in real time of topline COVID-19 outcomes

[1] Code for reproducing our experiments is available at https://github.com/tiffanyding/multiQT.

collected over the pandemic, in a large collaborative effort between researchers and the United States Center for Disease Control and Prevention (Cramer et al., 2022b). The COVID-19 Forecast Hub ran from April 2020 through April 2024, and it collected forecasts of COVID-19 cases, hospitalizations, and deaths for subsets of this full four-year period, at varying spatial and temporal resolutions. For our analysis, we focus on forecasts of weekly COVID-19 deaths at the state level, which were collected for 23 quantile levels,

$$\mathcal{A} = \{0.01, 0.025, 0.05, 0.1, 0.15, \ldots, 0.85, 0.9, 0.95, 0.975, 0.99\},$$

and at h weeks ahead, for horizons (lead times) $h \in \{1, 2, 3, 4\}$.

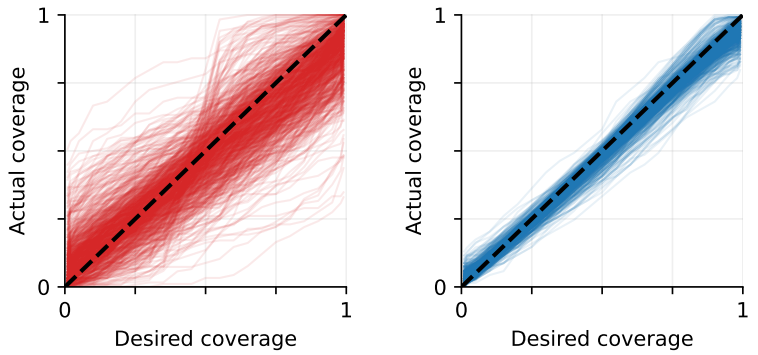
We apply our MultiQT procedure to weekly state-level COVID-19 death forecasts generated by 15 forecasting teams, corresponding to $15 \times 50 = 750$ time series for each forecast horizon h. This set of forecasters is obtained by starting from the set of forecasters considered in Cramer et al. (2022a), then filtering out forecasters with missing forecasts or forecasts for fewer than 50 time steps for any state. The selected forecasters have forecasts for periods ranging from 68 to 152 weeks.

We apply MultiQT separately to each forecaster-state combination. When applying MultiQT to forecasts that are $h = 1$ week ahead, we use the implementation described in Procedure 1; for $h \in \{2, 3, 4\}$, we run the delayed feedback version of MultiQT with the modification in (6), where $D = h - 1$. For some forecasting teams, their forecasts are well calibrated to begin with, while for others, their forecasts are systematically biased in some way (too low or too high, or their confidence bands are too narrow or too wide). In general, we find that wrapping the MultiQT method around these base forecasts successfully corrects such biases and improves calibration, as we now describe.

Figure 4a displays the calibration of one-week-ahead death forecasts from the COVID-19 Forecast Hub. Each colored line corresponds to a single forecaster for a single location. When one of these lines is below the black dashed line, it means that forecasts are biased downward for the corresponding levels, whereas being above the dashed line means that forecasts are biased upward. Both forms of miscalibration generally dilute the utility of forecasts to decision makers. Figure 4b plots the calibration of the same forecasts after applying MultiQT. We see that MultiQT reduces both types of bias (it brings lines closer to the diagonal, from above and below). Furthermore, Figure 14 in the appendix shows that MultiQT similarly improves two-, three-, and four-week-ahead forecasts.

Figure 5 illustrates the change in calibration error and quantile loss induced by MultiQT, with one panel per forecast horizon h. Each arrow represents one forecaster averaged over all states. The tail of each arrow represents the performance of the raw forecasts, while its head represents the performance after we apply MultiQT. All arrows point downward, which tells us that MultiQT achieves its goal of improving calibration. In fact, after recalibrating with MultiQT, most forecasters achieve calibration error close to zero, corresponding to perfect calibration. We also see that this improvement in calibration does not significantly degrade the quantile loss and, in fact, often leads to a slight improvement. This is consistent with the regret guarantee stated in Theorem 3 (and we note that our choice of learning rate in this section is more aligned with improving calibration, whereas if we were to target regret, we would choose a learning rate decreasing with time).

In Figures 15-18 in the appendix, we provide visualizations of the MultiQT-adjusted forecasts for each individual COVID-19 forecaster.



(a) Raw forecasts (b) MultiQT forecasts

Figure 4: Actual versus desired coverage for one-week-ahead COVID-19 death forecasts before (left) and after (right) applying MultiQT. Each line corresponds to a single forecaster for a single location.

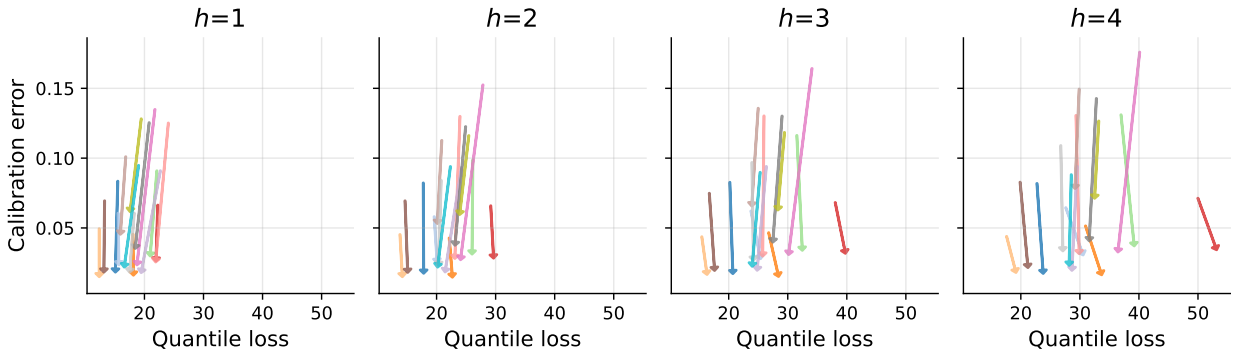
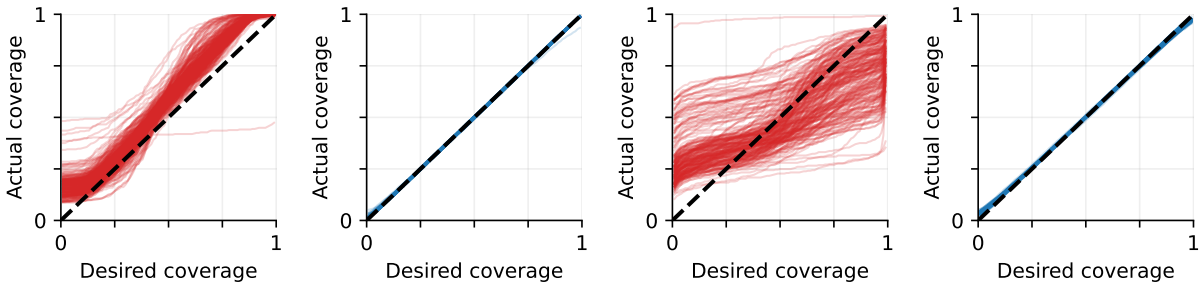


Figure 5: Calibration error versus quantile loss for raw forecasts (tail of arrow) and MultiQT forecasts (head) for h-week-ahead COVID-19 death forecasts, where $h \in \{1, 2, 3, 4\}$. Each color represents a forecaster, and the coordinates of the head and tail are determined by averaging the given metric across all 50 states for the specified horizon. For each metric, lower is better.

5.2 Energy forecasting

While renewable energy sources such as wind and solar hold great promise for reducing carbon emissions, a significant downside is that they suffer from uncertain production due to the inherent stochasticity of weather. This uncertainty must be properly accounted for in order to successfully integrate renewable energy sources into the energy grid. Grid operators rely on accurate forecasts of renewable energy production to determine whether (and for what times) it is necessary to procure additional energy reserves via what are known as balancing capacity markets (Hirth and Ziegenhagen, 2015; Regelleistung, 2024).

The ARPA-E PERFORM dataset was assembled to help develop more efficient and reliable energy grids (Bryce et al., 2023). It consists of probabilistic forecasts made by the National Renewable Energy Laboratory, a national laboratory of the U.S. Department of Energy, for wind and solar energy generated at various sites in the United States along with the actual values, all measured in megawatts. Quantile forecasts are made at 99 levels $\mathcal{A}$, which are evenly spaced from 0.1 to 0.99. For our analysis, we focus on day-ahead forecasting for sites belonging to the Electric Reliability Council of Texas (ERCOT), the main operator of the electrical



(a) Wind (b) Solar

Figure 6: Actual versus desired coverage for day-ahead wind (a) and solar (b) energy forecasts for the 10:00 a.m. time period. In each of (a) and (b), the left panel corresponds to raw forecasts, and the right panel to MultiQT-adjusted forecasts; each line corresponds to a different site.

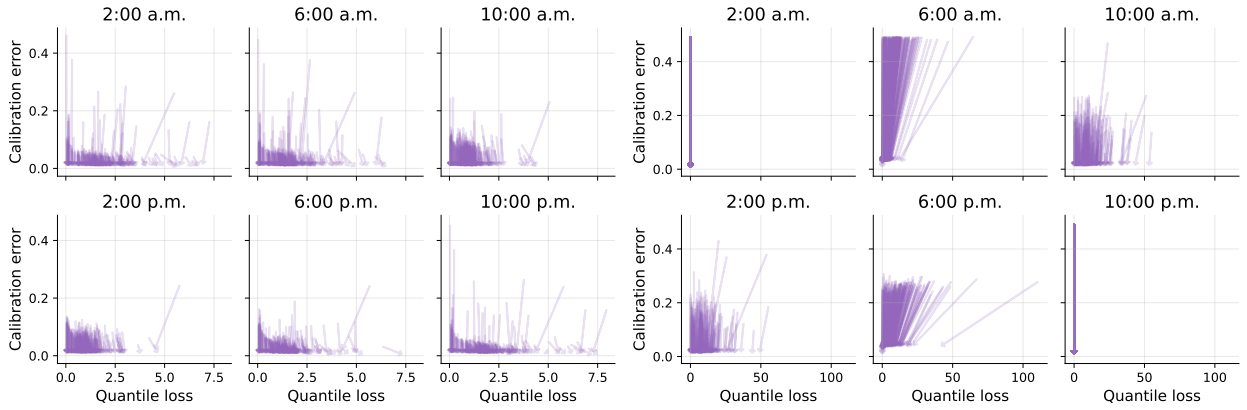
grid in Texas. For wind power, there are 264 sites, and for solar power, there are 226 proposed sites, making a total of 490 sites. Day-ahead forecasts are made at 12:00 p.m. CST each day for energy production during each hour of the subsequent day. The dataset provides forecasts for each day of 2018.

We run MultiQT separately for each hour of the day. For example, one sequence of targets we consider is the wind production of a particular site at 10:00 a.m. on January 1, 10:00 a.m. on January 2, 10:00 a.m. on January 3, and so on. Motivated by how balancing capacity products are available in four-hour blocks (Regelleistung, 2024), we specifically focus on the hours 2:00 a.m., 6:00 a.m., 10:00 a.m., 2:00 p.m., 6:00 p.m., and 10:00 p.m. Each of these hours belongs to a different four-hour block and its forecasts can be used to inform whether a balancing capacity product will be needed for that time block. For the first three hours we consider (2:00 a.m., 6:00 a.m., 10:00 a.m.), feedback from the previous day's forecast is available before the next day's forecasts are issued at 12:00 p.m., so there is no delay in feedback. However, for the afternoon and evening hours (2:00 p.m., 6:00 p.m., and 10:00 p.m.), there is a one-day delay because we do not observe feedback for these hours before issuing the next day's forecasts. Therefore, for these hours, we run MultiQT with a feedback delay of $D = 1$.

Figure 6 displays the calibration of quantile forecasts before and after applying MultiQT to the forecasts for energy production at 10:00 a.m. We can see that the raw wind forecasts are generally biased upward, with calibration curves falling above the diagonal line, and the solar forecasts are generally too narrow, with calibration curves that are too flat (nearly horizontal). MultiQT corrects each of these issues and delivers near perfect calibration. A similar improvement in the calibration of energy forecasts can be seen for other hours of the day, displayed in Figure 19 in the appendix.

Figure 7 again illustrates the change in calibration error and quantile loss induced by MultiQT for all six hours we consider, in the same format as Figure 5 for the COVID-19 dataset. Here, each arrow corresponds to a different site. The results are qualitatively similar to those for the COVID-19 dataset: MultiQT consistently improves forecast calibration and never substantially increases the average quantile loss. In particular, for the solar forecasts, we generally see a strong improvement in quantile loss due to the extremely poor calibration of the raw forecasts.

As a case study to better understand how MultiQT changes the raw forecasts, we visualize the forecasts before and after applying MultiQT for a wind energy site whose raw forecasts were particularly miscalibrated in Figure 8. For visual clarity, we show forecasts only for a 50-day period starting from March 1, 2018 but



(a) Wind (b) Solar

Figure 7: Calibration error versus quantile loss for raw forecasts (tail of arrow) and MultiQT forecasts (head) for day-ahead wind and solar energy production at 2:00 a.m., 6:00 a.m., 10:00 a.m., 2:00 p.m., 6:00 p.m., and 10:00 p.m. Each arrow represents a different site. For each metric, lower is better.

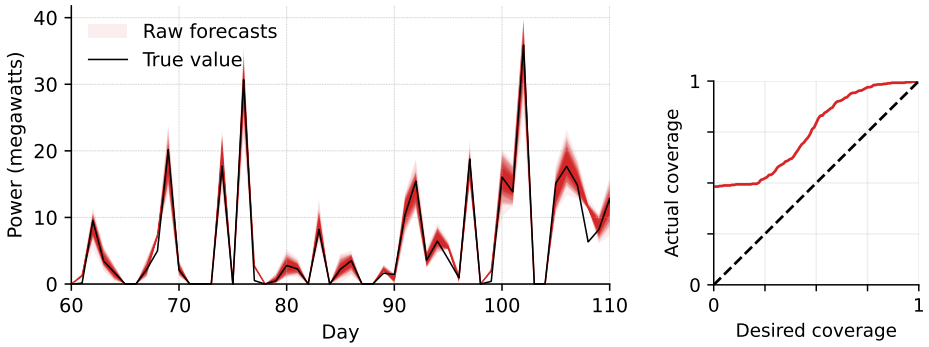
the underlying experiment covers the entire year of 2018. We can see that the raw forecasts are upwardly biased and too narrow in many places compared to the true energy output. MultiQT largely corrects for this and provides a better representation of the uncertainty. In Figures 21-22 in the appendix, we present analogous plots for additional wind and solar sites.

6 Discussion

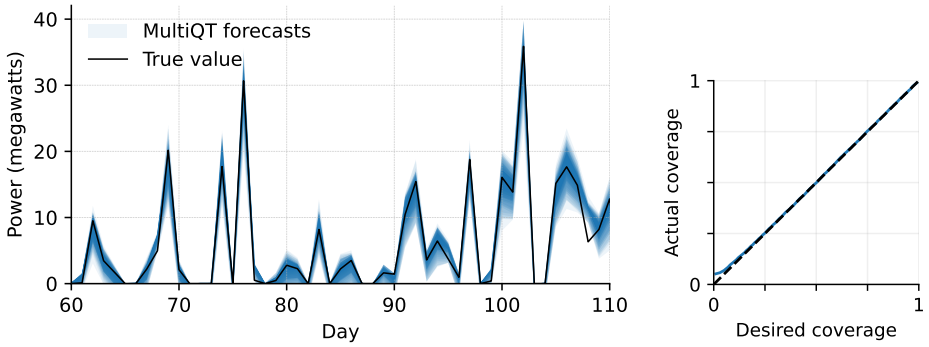
In this paper, we proposed a simple procedure which wraps around any existing online quantile forecaster to produce corrected forecasts that are guaranteed to be calibrated without crossing, meaning quantile forecasts at successive levels are always properly ordered. Our method, MultiQT, is an instance of lazy gradient descent applied to a particular online learning problem, involving quantile losses and the isotonic cone. To establish a calibration guarantee (for arbitrary and potentially even adversarial data sequences), we abstract to a more general problem of achieving constrained gradient equilibrium (GEQ) via lazy gradient descent, and we derive new gradient equilibrium theory for this algorithm. We also derive a regret guarantee with respect to the quantile loss. In experiments with datasets from COVID-19 and energy forecasting, we find that MultiQT significantly improves the calibration of real forecasters, for the most part without sacrificing quantile loss, and often slightly improving it.

We finish by discussing some topics related to the main thrust of our paper, and ideas for future work.

From quantile forecasts to prediction intervals. Throughout, we have touched on some reasons why achieving calibration while maintaining distributional consistency is an important problem. Here is yet another useful consequence: these two properties together allow us to construct nested prediction intervals with the correct long-run coverage, where "nested" means that the $(1-\alpha)$-level interval will always be contained in the $(1-\beta)$-level interval when $\alpha > \beta$. We emphasize that properly ordered quantiles is critical for obtaining nested intervals; otherwise nestedness may not be satisfied, e.g., at a given time step we might have one or both of the endpoints of the 0.5-level prediction interval lying outside the 0.9-level prediction



(a) Raw forecasts and their calibration.



(b) Forecasts and calibration after applying MultiQT.

Figure 8: Day-ahead wind energy forecasts for the site `Wind_Power_Partners_94_Wind_Farm` for 10:00 a.m. each day from March 1, 2018 to April 20, 2018, before (top) and after (bottom) applying MultiQT.

interval. Coverage of such intervals can be verified directly: to construct equi-tailed $(1-\alpha)$-level intervals, we can define $I_t^{1-\alpha} = (q_t^{\alpha/2}, q_t^{1-\alpha/2}]$, and then

$$\begin{aligned}\frac{1}{T}\sum_{t=1}^{T} \mathbb{1}\{y_t \in I_t^{1-\alpha}\} &= 1 - \frac{1}{T}\sum_{t=1}^{T} \mathbb{1}\{y_t \leq q_t^{\alpha/2}\} - \frac{1}{T}\sum_{t=1}^{T} \mathbb{1}\{y_t > q_t^{1-\alpha/2}\} \\ &= 1 - \alpha/2 - \alpha/2 = 1 - \alpha.\end{aligned}$$

Note that have implicitly relied on properly ordered quantiles in using $q_t^{\alpha/2} \leq q_t^{1-\alpha/2}$ for all t. Further, if each y_t is continuously distributed, then using closed intervals $[q_t^{\alpha/2}, q_t^{1-\alpha/2}]$, $t = 1, 2, \ldots$ will achieve the correct coverage with probability one.

Faster rates for GEQ and calibration error. In Section 4, we showed that the calibration error of the MultiQT forecasts approaches zero at a $1/\sqrt{T}$ rate. This was established based on new theory for constrained gradient equilibrium in Section 3. Here we refine these analyses to give faster rates under stronger assumptions. Proofs are given in Appendix D.

We first refine the gradient equilibrium result in Proposition 5 by assuming that the function ϕ that appears in the definition of restorativity (15) can be lower bounded by a positive constant, and that the distance between pairs of hidden and played iterates remains bounded.

Proposition 7. *Under the conditions of Proposition 5, additionally assume that each loss ℓ_t is now (h_t, ϕ_t)-restorative, with $\phi_t(\theta) \geq \eta L^2/2$ for $\|\theta\|_2 > h_t$, and that each pair of hidden and played iterates $(\tilde{\theta}_t, \theta_t)$ remains within a bounded distance of each other: $\|\tilde{\theta}_t - \theta_t\|_2 \leq B$. If h_t is a nondecreasing sequence, then, for all $T \geq 1$, the lazy gradient descent iterates in (17) and (18) satisfy*

$$\|\tilde{\theta}_{T+1}\|_2 \leq \max\{\|\tilde{\theta}_1\|_2, h_T\} + B + \eta L.$$

This implies

$$\left\| \frac{1}{T} \sum_{t=1}^{T} g_t(\theta_t) \right\|_2 \leq \frac{2\|\tilde{\theta}_1\|_2}{\eta T} + \frac{L}{T} + \frac{h_T + B}{\eta T},$$

which goes to zero as $T \to \infty$ as long as h_T is sublinear in T.

When $h_t = h$ for all t, Proposition 5 gives a $1/\sqrt{T}$ rate for gradient equilibrium. By adding a stronger assumption on ϕ, and importantly, the assumption that hidden and played sequences do not diverge away from each other, we see that Proposition 7 improves this to a $1/T$ rate.

The assumption that ϕ is lower bounded by a positive constant, which Angelopoulos et al. (2025) refer to as a "positive curvature" condition, is not strong. We can show that quantile loss satisfies this condition as an extension of Lemma 1. The assumption that $\|\tilde{\theta}_t - \theta_t\|_2$ remains bounded is trickier to analyze. Though it seems intuitive that in most instances we would expect this to be the case, it is nonetheless challenging to verify formally. Fortunately, the next lemma shows that this is true for MultiQT in a specialized setting, where the base forecasts are point forecasts.

Lemma 3. *If the base forecasts are point forecasts, i.e., $b_t^\alpha = \mu_t$ for all α and t, and $|y_t - \mu_t| \leq R$ for all t, then the MultiQT iterates starting from initialization $\tilde{\theta}_1 \in \mathcal{K}$ satisfy $\|\tilde{\theta}_t - \theta_t\|_2 \leq \eta m^{3/2}/\sqrt{3}$ for all t.*

Combining the previous results we get the following $1/T$ rate on the calibration error of MultiQT when the base forecasts are point forecasts.

Corollary 1. *Under the conditions of Lemma 3, for all $T \geq 1$, the MultiQT iterates from Procedure 1 obey the ℓ_2 coverage error bound*

$$\sqrt{\sum_{\alpha \in \mathcal{A}} \left(\frac{1}{T} \sum_{t=1}^{T} \mathrm{cov}_t^\alpha - \alpha \right)^2} \leq \frac{2\|\tilde{\theta}_1\|_2}{\eta T} + \frac{\sqrt{m}}{T} + \frac{m^{3/2}}{2 d_{\mathcal{A}} T} + \frac{R m^{3/2}}{d_{\mathcal{A}} \eta T} + \frac{m^{3/2}}{\sqrt{3} T},$$

where recall $m = |\mathcal{A}|$, and $d_{\mathcal{A}}$ is as in Lemma 1. Additionally recall that, since $\|x\|_\infty \leq \|x\|_2$ for any vector $x \in \mathbb{R}^m$, the right-hand side above is also an upper bound for $\left| \frac{1}{T} \sum_{t=1}^{T} \mathrm{cov}_t^\alpha - \alpha \right|$, for each $\alpha \in \mathcal{A}$.

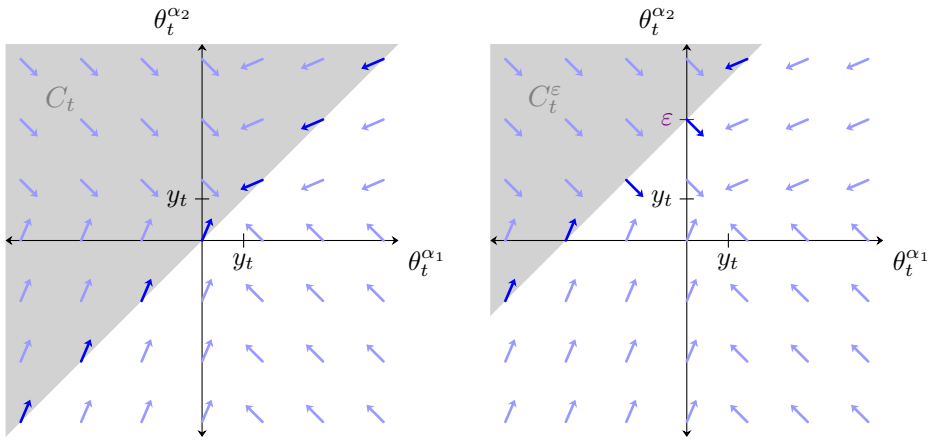
It is an open question whether MultiQT maintains a bounded distance between hidden and played iterates in general, when each b_t is an arbitrary ordered vector of quantile forecasts, and hence whether the $1/T$ rate of Corollary 1 extends to this general setting.

Inward flow. A key condition we used in our analysis is inward flow, which says that the negative gradient field points inwards at the boundary of the constraint set. We showed that lazy gradient descent achieves constrained gradient equilibrium when inward flow is satisfied, and our calibration guarantee for MultiQT relied on the fact that the MultiQT loss and constraint set jointly satisfy inward flow. This property is far from trivial, and can fail even in seemingly natural modifications of the MultiQT problem; for example, we might seek ε-separated quantiles, and define for a constant $\varepsilon \geq 0$ the constraint set

$$C_t^\varepsilon = \left\{ x \in \mathbb{R}^m : x_i + \varepsilon \leq x_{i+1},\ i = 1, 2, \ldots, m-1 \right\} - b_t. \tag{25}$$

Notice that setting $\varepsilon = 0$ recovers the original constraint set (12). Unfortunately, the MultiQT loss and the ε-separated constraint set C_t^ε do not satisfy inward flow. This is visualized in Figure 9b: we can see that the negative gradient, denoted by the small arrows, does not point inwards at all points on the boundary of C_t^ε. We can contrast this with Figure 9a, which visualizes the constraint used in MultiQT.

Violating inward flow means that the result in Theorem 1 does not apply but leaves open the possibility that other analyses may be used to establish a calibration result; however, we show that this is not the case for the ε-separated constraint set, and construct a formal negative example in Appendix C.



(a) Original constraint set

(b) ε-separated constraint set

Figure 9: Visualization of the negative gradient field of the MultiQT loss (arrows) for two quantiles, with no base forecaster ($b_t^{\alpha_1} = b_t^{\alpha_2} = 0$), and with the target value y_t as drawn. The inward flow property is satisfied for the original MultiQT constraint set (left), but not the ε-separated constraint set (right).

In general, it is unclear to us to what degree inward flow is necessary for lazy gradient descent to achieve constrained gradient equilibrium, and if not, whether there are other sufficient conditions that may be more naturally satisfied in some settings. The study of gradient equilibrium under iterate constraints is still nascent and requires further development.

Future work. We close with some ideas for future work, in addition to ones already mentioned. In multi-horizon forecasting settings, such as the COVID-19 forecasting problem (where forecasts were simultaneously issued for outcomes one, two, three, and four weeks ahead), the forecast residuals share correlation between successive horizons, and one may leverage scorecasting techniques as in Angelopoulos et al. (2023); Wang and Hyndman (2024) to improve sharpness at an individual quantile level. Combining this with MultiQT would be an important practical development.

Another idea is to approach conditional notions of calibration, which require coverage to be obtained at each quantile level conditional on the quantile prediction between issued. This is of course stronger than the notion considered in our paper (which does not perform any conditioning). It is worth noting that our notion of coverage in this paper corresponds to a discretization of what is called probabilistic calibration (also called PIT calibration) in the forecasting literature, see, e.g., Gneiting et al. (2007). Conditional versions will have different names, depending on what precisely is being conditioned on, with the strongest version being called auto-calibration, see, e.g., Gneiting and Resin (2023). As a future direction, it would be desirable to be able

to encode conditional notions of calibration as a form of gradient equilibrium, and show this can be obtained with standard lightweight online methods such as (lazy) gradient descent, as existing methods for achieving conditional calibration such as Noarov et al. (2023) are more computationally complex.

A final idea would be to consider an infinite-dimensional version of our problem, where the base forecasts take the form of a quantile function $b_t : [0, 1] \to \mathbb{R}$. This could be seen as taking $m \to \infty$ in our current setup. This poses numerous challenges (algorithmically and theoretically), but would nonetheless be an interesting direction.

Acknowledgments

We thank Rina Barber and Aaron Roth for helpful discussions and Erez Buchweitz for guidance on working with the COVID-19 Forecast Hub dataset. TD was supported by the National Science Foundation Graduate Research Fellowship Program grant no. 2146752. IG and RJT were supported by the Office of Naval Research grant no. N00014-20-1-2787.

References

Anastasios N. Angelopoulos, Emmanuel J. Candès, and Ryan J. Tibshirani. Conformal PID control for time series prediction. In *Advances in Neural Information Processing Systems*, 2023.

Anastasios N. Angelopoulos, Michael I. Jordan, and Ryan J. Tibshirani. Gradient equilibrium in online learning: Theory and applications. arXiv: 2501.08330, 2025.

Miriam Ayer, H. Daniel Brunk, George M. Ewing, William T. Reid, and Edward Silverman. An empirical distribution function for sampling with incomplete information. *Annals of Mathematical Statistics*, 26(4): 641–647, 1955.

Richard E. Barlow, D. J. Bartholomew, J. M. Bremner, and H. D. Brunk. *Statistical Inference Under Order Restrictions: The Theory and Application of Isotonic Regression*. Wiley, 1972.

Gilbert Bassett and Roger Koenker. An empirical quantile function for linear models with iid errors. *Journal of the American Statistical Association*, 77(378):407–415, 1982.

Osbert Bastani, Varun Gupta, Christopher Jung, Georgy Noarov, Ramya Ramalingam, and Aaron Roth. Practical adversarial multivalid conformal prediction. In *Advances in Neural Information Processing Systems*, 2022.

Aadyot Bhatnagar, Huan Wang, Caiming Xiong, and Yu Bai. Improved online conformal prediction via strongly adaptive online learning. In *Proceedings of the International Conference on Machine Learning*, 2023.

Johannes Bracher, Evan L. Ray, Tilmann Gneiting, and Nicholas G. Reich. Evaluating epidemic forecasts in an interval format. *PLOS Computational Biology*, 17(2):1–15, 2021.

Sarah Brocklehurst, P. Y. Chan, Bev Littlewood, and John Snell. Recalibrating software reliability models. *IEEE Transactions on Software Engineering*, 16(4):458–470, 1990.

Richard Bryce, Grant Buster, Kate Doubleday, Cong Feng, Ross Ring-Jarvi, Michael Rossol, Flora Zhang, and Bri-Mathias Hodge. Solar PV, wind generation, and load forecasting dataset for ERCOT 2018: Performance-based energy resource feedback, optimization, and risk management (PERFORM). National Renewable Energy Laboratory Tech Report, 2023.

Erez Buchweitz, João Vitor Romano, and Ryan J. Tibshirani. Asymmetric penalties underlie proper loss functions in probabilistic forecasting. arXiv: 2505.00937, 2025.

Ying Cao and Zuo-Jun Max Shen. Quantile forecasting and data-driven inventory management under nonstationary demand. *Operations Research Letters*, 47(6):465–472, 2019.

Victor Chernozhukov, Iván Fernández-Val, and Alfred Galichon. Quantile and probability curves without crossing. *Econometrica*, 78(3):1093–1125, 2010.

Estee Y. Cramer et al. Evaluation of individual and ensemble probabilistic forecasts of COVID-19 mortality in the United States. *Proceedings of the National Academy of Sciences*, 119(15):e2113561119, 2022a.

Estee Y. Cramer et al. The United States COVID-19 Forecast Hub dataset. *Scientific Data*, 9, 2022b.

Shachi Deshpande, Charles Marx, and Volodymyr Kuleshov. Calibrated regression against an adversary without regret. arXiv: 2302.12196, 2023.

Colin Doms, Sarah C. Kramer, and Jeffrey Shaman. Assessing the use of influenza forecasts and epidemiological modeling in public health decision making in the United States. *Scientific Reports*, 8(1):12406, 2018.

Rasool Fakoor, Taesup Kim, Jonas Mueller, Alexander Smola, and Ryan J. Tibshirani. Flexible model aggregation for quantile regression. *Journal of Machine Learning Research*, 24(162):1–45, 2023.

Shai Feldman, Liran Ringel, Stephen Bates, and Yaniv Romano. Achieving risk control in online learning settings. arXiv: 2205.09095, 2022.

Dean P. Foster. A proof of calibration via Blackwell's approachability theorem. *Games and Economic Behavior*, 29(1-2):73–78, 1999.

Dean P. Foster and Sergiu Hart. Forecast hedging and calibration. *Journal of Political Economy*, 129(12): 3447–3490, 2021.

Jan Gasthaus, Konstantinos Benidis, Yuyang Wang, Syama Sundar Rangapuram, David Salinas, Valentin Flunkert, and Tim Januschowski. Probabilistic forecasting with spline quantile function RNNs. In *Proceedings of the International Conference on Artificial Intelligence and Statistics*, 2019.

Isaac Gibbs and Emmanuel J. Candès. Adaptive conformal inference under distribution shift. In *Advances in Neural Information Processing Systems*, 2021.

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

Tilmann Gneiting and Adrian E. Raftery. Strictly proper scoring rules, prediction, and estimation. *Journal of the American Statistical Association*, 102(477):359–378, 2007.

Tilmann Gneiting and Roopesh Ranjan. Combining predictive distributions. *Electronic Journal of Statistics*, 7:1747–1782, 2013.

Tilmann Gneiting and Johannes Resin. Regression diagnostics meets forecast evaluation: Conditional calibration, reliability diagrams, and coefficient of determination. *Electronic Journal of Statistics*, 17(2): 3226–3286, 2023.

Tilmann Gneiting, Fadoua Balabdaoui, and Adrian E. Raftery. Probabilistic forecasts, calibration and sharpness. *Journal of the Royal Statistical Society: Series B*, 69(2):243–268, 2007.

Tilmann Gneiting, Daniel Wolffram, Johannes Resin, Kristof Kraus, Johannes Bracher, Timo Dimitriadis, Veit Hagenmeyer, Alexander I. Jordan, Sebastian Lerch, Kaleb Phipps, et al. Model diagnostics and forecast evaluation for quantiles. *Annual Review of Statistics and Its Application*, 10(1):597–621, 2023.

Varun Gupta, Christopher Jung, Georgy Noarov, Mallesh M. Pai, and Aaron Roth. Online multivalid learning: Means, moments, and prediction intervals. In *Innovations in Theoretical Computer Science Conference*, 2022.

Erfan Hajihashemi and Yanning Shen. Multi-model ensemble conformal prediction in dynamic environments. In *Advances in Neural Information Processing Systems*, 2024.

Thomas M Hamill and Stephen J. Colucci. Verification of Eta–RSM short-range ensemble forecasts. *Monthly Weather Review*, 125(6):1312–1327, 1997.

Elad Hazan. Introduction to online convex optimization. arXiv: 1909.05207, 2019.

Lion Hirth and Inka Ziegenhagen. Balancing power and variable renewables: Three links. *Renewable and Sustainable Energy Reviews*, 50:1035–1051, 2015.

Tao Hong and Shu Fan. Probabilistic electric load forecasting: A tutorial review. *International Journal of Forecasting*, 32(3):914–938, 2016.

Dongjian Hu, Junxi Wu, Shu-Tao Xia, and Changliang Zou. Distribution-informed online conformal prediction. arXiv: 2512.07770, 2025.

Roger Koenker and Gilbert Bassett. Regression quantiles. *Econometrica*, 46(1):33–50, 1978.

Jordan Lekeufack, Anastasios N Angelopoulos, Andrea Bajcsy, Michael I Jordan, and Jitendra Malik. Conformal decision theory: Safe autonomous decisions from imperfect predictions. In *Proceedings of the International Conference on Robotics and Automation*, 2024.

Ruipu Li and Alexander Rodríguez. Neural conformal control for time series forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2025.

Yufeng Liu and Yichao Wu. Stepwise multiple quantile regression estimation using non-crossing constraints. *Statistics and Its Interface*, 2(3):299–310, 2009.

Chelsea S. Lutz, Mimi P. Huynh, Monica Schroeder, Sophia Anyatonwu, F. Scott Dahlgren, Gregory Danyluk, Danielle Fernandez, Sharon K. Greene, Nodar Kipshidze, et al. Applying infectious disease forecasting to public health: A path forward using influenza forecasting examples. *BMC Public Health*, 19(1):1659, 2019.

Charles Marx, Volodymyr Kuleshov, and Stefano Ermon. Calibrated probabilistic forecasts for arbitrary sequences. arXiv: 2409.19157, 2024.

H. Brendan McMahan. A survey of algorithms and analysis for adaptive online learning. *Journal of Machine Learning Research*, 18:1–50, 2017.

Georgy Noarov, Ramya Ramalingam, Aaron Roth, and Stephan Xie. High-dimensional prediction for sequential decision making. arXiv: 2310.17651, 2023.

Youngsuk Park, Danielle Maddix, François-Xavier Aubet, Kelvin Kan, Jan Gasthaus, and Yuyang Wang. Learning quantile functions without quantile crossing for distribution-free time series forecasting. In *Proceedings of the International Conference on Artificial Intelligence and Statistics*, 2022.

Juan Carlos Perdomo and Benjamin Recht. In defense of defensive forecasting. arXiv: 2506.11848, 2025.

Adrian E. Raftery, Tilmann Gneiting, Fadoua Balabdaoui, and Michael Polakowski. Using Bayesian model averaging to calibrate forecast ensembles. *Monthly Weather Review*, 133(5):1155–1174, 2005.

Regelleistung. How does the balancing market work?, 2024. URL https://www.regelleistung.net/en-us/Basics-of-balancing-services/How-does-the-balancing-market-work.

Aaron Rumack, Ryan J. Tibshirani, and Roni Rosenfeld. Recalibrating probabilistic forecasts of epidemics. *PLOS Computational Biology*, 18(12):e1010771, 2022.

Shai Shalev-Shwartz. Online learning and online convex optimization. *Foundations and Trends in Machine Learning*, 4(2):107–194, 2012.

Huug van den Dool, Emily Becker, Li-Chuan Chen, and Qin Zhang. The probability anomaly correlation and calibration of probabilistic forecasts. *Weather and Forecasting*, 32(1):199–206, 2017.

Vladimir Vovk, Akimichi Takemura, and Glenn Shafer. Defensive forecasting. In *International Workshop on Artificial Intelligence and Statistics*, 2005.

Xiaoqian Wang and Rob J. Hyndman. Online conformal inference for multi-step time series forecasting. arXiv: 2410.13115, 2024.

Zitong Yang, Emmanuel J. Candès, and Lihua Lei. Bellman conformal inference: Calibrating prediction intervals for time series. arXiv: 2402.05203, 2024.

Margaux Zaffran, Olivier Féron, Yannig Goude, Julie Josse, and Aymeric Dieuleveut. Adaptive conformal predictions for time series. In *Proceedings of the International Conference on Machine Learning*, 2022.

Zhiyu Zhang, Zhou Lu, and Heng Yang. The benefit of being Bayesian in online conformal prediction. arXiv: 2410.02561, 2024.

A Distributional inconsistency of QT

To demonstrate that running the quantile tracker (QT) separately for each quantile level cannot be used to solve calibration without crossings, we run this method on the COVID-19 dataset from Cramer et al. (2022b), as described in Section 5. We also use the learning rate heuristic described in that section. We run QT on 750 time series of one-week-ahead forecasts of weekly COVID-19 deaths at the state level (15 forecasters $\times$ 50 states). Figure 10 plots the fraction of time steps in each time series that have at least one pair of crossed quantiles (where we say a crossing has occurred at time t if there exists quantile levels $\alpha < \beta$ where the corresponding QT-adjusted forecasts satisfy $q_t^\alpha > q_t^\beta$). We see QT produces distributionally inconsistent quantiles at 87% of time steps on average, which is practically undesirable.

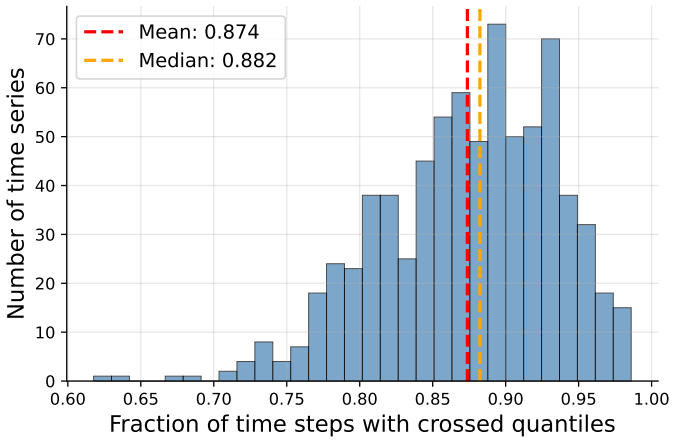


Figure 10: Histogram of the fraction of time steps with crossings after applying QT to one-week-ahead COVID-19 death forecasts separately for each quantile, over the 15 forecasting teams and all 50 states.

B Proofs for Sections 3 and 4

We prove results for the delayed feedback setting with a constant delay of $D \geq 0$. Results for the no-delay setting follow immediately by setting $D = 0$. The object of our analysis is lazy gradient descent (lazy GD) with constant delay, which we now discuss in some further detail.

With delay $D \geq 0$, we do not observe y_t at time t (except when $D = 0$); rather, it is observed after we play our action at time $t + D$, at which point we use it to take a gradient step. Recall that lazy gradient descent in the current setting is given by (21) followed by (17), where for convenience we set $g_t(\theta_t) = 0$ for $t \leq 0$. By unrolling (21) we obtain

$$\tilde{\theta}_{t+1} = \tilde{\theta}_1 - \eta \sum_{s=1}^{t-D} g_s(\theta_s), \tag{26}$$

where we adopt the convention that the summation from a to b is zero if $b \leq a$. Using (26), we get two simple facts that we will use below. First, substituting (26) into (17) allows us to rewrite the whole lazy gradient descent algorithm as

$$\theta_{t+1} = \Pi_{C_{t+1}}\left(\tilde{\theta}_1 - \eta \sum_{s=1}^{t-D} g_s(\theta_s)\right). \tag{27}$$

Second, by rearranging (26) and using the triangle inequality, we get that the average gradient satisfies

$$\left\| \frac{1}{T} \sum_{t=1}^{T} g_t(\theta_t) \right\|_2 \leq \frac{\|\tilde{\theta}_1\|_2 + \|\tilde{\theta}_{T+D+1}\|_2}{\eta T}. \tag{28}$$

B.1 Constrained gradient equilibrium for lazy gradient descent with delay

In this section, we prove the results needed to show that lazy gradient descent (with delay) achieves constrained gradient equilibrium when inward flow holds.

B.1.1 Proof of Proposition 6

We follow the general proof structure from Proposition 5 of Angelopoulos et al. (2025). We begin by expanding the square in (18):

$$\begin{aligned} \|\tilde{\theta}_{T+D+1}\|_2^2 &= \|\tilde{\theta}_{T+D}\|_2^2 + \eta^2 \|g_T(\theta_T)\|_2^2 - 2\eta \langle g_T(\theta_T), \tilde{\theta}_{T+D} \rangle \\ &\leq \|\tilde{\theta}_{T+D}\|_2^2 + \eta^2 L^2 - 2\eta \langle g_T(\theta_T), \tilde{\theta}_{T+D} \rangle, \end{aligned} \tag{29}$$

where the second line uses Lipschitzness. We focus on bounding the third term in (29), which we rewrite as

$$-2\eta \langle g_T(\theta_T), \tilde{\theta}_{T+D} \rangle = -2\eta \langle g_T(\theta_T), \tilde{\theta}_T \rangle - 2\eta \langle g_T(\theta_T), \tilde{\theta}_{T+D} - \tilde{\theta}_T \rangle. \tag{30}$$

We start by bounding the first term in (30). Due to inward flow, instead of bounding the inner product of the gradient and the hidden iterate, we can instead bound the inner product with the played iterate: by part (ii) of Lemma 4 below, we have $-\langle g_T(\theta_T), \tilde{\theta}_T \rangle \leq -\langle g_T(\theta_T), \theta_T \rangle$. Proceeding in cases, if $\|\theta_T\|_2 > h_T$, then the restorativity condition kicks in and we have

$$-\langle g_T(\theta_T), \theta_T \rangle \leq 0.$$

Otherwise, if $\|\theta_T\|_2 \leq h_T$, then we have

$$-\langle g_T(\theta_T), \theta_T \rangle \leq \|g_T(\theta_T)\|_2 \|\theta_T\|_2 \leq L h_T$$

by Cauchy-Schwarz, Lipschitzness, and the assumption on $\|\theta_T\|_2$. Combining the above arguments, we have $-\langle g_T(\theta_T), \tilde{\theta}_T \rangle \leq \max\{0, L h_T\} = L h_T$.

The second term in (30) is the penalty we incur for delayed feedback. To bound it, note that

$$\begin{aligned} -\langle g_T(\theta_T), \tilde{\theta}_{T+D} - \tilde{\theta}_T \rangle &\leq \|g_T(\theta_T)\|_2 \|\tilde{\theta}_{T+D} - \tilde{\theta}_T\|_2 \\ &\leq \|g_T(\theta_T)\|_2 \left\| \eta \sum_{t=T}^{T+D-1} g_t(\theta_t) \right\|_2 \\ &\leq \eta \|g_T(\theta_T)\|_2 \left(\sum_{t=T}^{T+D-1} \|g_t(\theta_t)\|_2 \right) \\ &\leq \eta D L^2, \end{aligned}$$

where the first line uses Cauchy-Schwarz, the second line is due to (21), the third uses the triangle inequality, and the fourth uses the Lipschitzness assumption.

Inserting both of these bounds into (30), we get

$$-2\eta \langle g_T(\theta_T), \tilde{\theta}_{T+D} \rangle \leq 2\eta L h_T + 2\eta^2 D L^2.$$

Plugging this back into (29), we obtain

$$\begin{aligned}\|\tilde{\theta}_{T+D+1}\|_2^2 &\le \|\tilde{\theta}_{T+D}\|_2^2 + \eta^2 L^2 + 2\eta L h_T + 2\eta^2 D L^2 \\ &= \|\tilde{\theta}_{T+D}\|_2^2 + \eta^2 L^2 (2D+1) + 2\eta L h_T \\ &\le \|\tilde{\theta}_{D+1}\|_2^2 + \eta^2 L^2 (2D+1) T + 2\eta L \sum_{t=1}^{T} h_t \\ &\le \|\tilde{\theta}_1\|_2^2 + \eta^2 L^2 (2D+1) T + 2\eta L \sum_{t=1}^{T} h_t,\end{aligned}$$

where the last line uses $\|\tilde{\theta}_t\|_2 = \|\tilde{\theta}_1\|_2$, for all $t \le D+1$. Taking a square root gives the bound on $\|\tilde{\theta}_{T+D+1}\|_2$ stated in the theorem.

To get the bound on the ℓ_2 norm of the average gradient, we first simplify the $\|\tilde{\theta}_{T+D+1}\|_2$ bound by observing that the nondecreasing property of h_t implies $\sum_{t=1}^{T} h_t \le T h_T$. We then apply the fact $\sqrt{a+b} \le \sqrt{a} + \sqrt{b}$, and lastly invoke (28).

B.1.2 Gradient alignment lemma

We now state and prove a fact used in the previous proof about the effect of projection on the inner product of an iterate and its gradient when inward flow is satisfied. We use $N_C(x) = \{v : v^\mathsf{T}(x-y) \ge 0 \text{ for all } y \in C\}$ to denote the normal cone of a set C at x.

Lemma 4. *If a loss ℓ (with gradient g) and a closed convex set C satisfy inward flow, then:*

(i) $\langle v, g(z)\rangle \ge 0$ *for all* $z \in \mathrm{bd}(C)$ *and* $v \in N_C(z)$.

(ii) $\langle z, g(\Pi_C(z))\rangle \ge \langle \Pi_C(z), g(\Pi_C(z))\rangle$ *for all* z.

Proof. For part (i), by definition of inward flow, we know that there exists $\varepsilon > 0$ such that $z - \varepsilon g(z) = \omega$ for some $\omega \in C$. We can thus write

$$\begin{aligned}\langle v, g(z)\rangle &= \frac{1}{\varepsilon}\langle v, \varepsilon g(z)\rangle \\ &= \frac{1}{\varepsilon}\langle v, z - \omega\rangle \\ &\ge 0,\end{aligned}$$

where the inequality holds as $v \in N_C(z)$. For part (ii), if $z \in C$, the result is trivial. Now consider $z \notin C$. Abbreviating $z_0 = \Pi_C(z)$, by definition of Euclidean projection, there exists $v \in N_C(z_0)$ such that $z = z_0 + v$. As $\langle v, g(z_0)\rangle \ge 0$ by part (i), it follows that $\langle z, g(z_0)\rangle = \langle z_0, g(z_0)\rangle + \langle v, g(z_0)\rangle \ge \langle z_0, g(z_0)\rangle$, and this completes the proof. □

B.2 Calibration theory for MultiQT

What remains now is to prove Lemmas 1 and 2, which we do below.

B.2.1 Proof of Lemma 1

Suppose $\|\theta_t\|_2 > h$. This implies there exists $\alpha^* \in \mathcal{A}$ such that $|\theta_t^{\alpha^*}| > h/\sqrt{m}$, because otherwise we would have $\|\theta_t\|_2^2 = \sum_{i=1}^m (\theta_t^{\alpha_i})^2 \leq \sum_{i=1}^m h^2/m = h^2$. Now expand the inner product:

$$\begin{aligned}\langle \theta_t, g_t(\theta_t)\rangle &= \sum_{\alpha\in\mathcal{A}} \theta_t^\alpha(\mathrm{cov}_t^\alpha - \alpha) \\ &= \sum_{\alpha:\theta_t^\alpha<-R} \theta_t^\alpha(\mathrm{cov}_t^\alpha - \alpha) + \sum_{\alpha:\theta_t^\alpha>R} \theta_t^\alpha(\mathrm{cov}_t^\alpha - \alpha) + \sum_{\alpha:-R\leq\theta_t^\alpha\leq R} \theta_t^\alpha(\mathrm{cov}_t^\alpha - \alpha). \end{aligned} \tag{31}$$

We will show that the first two sums must be positive and then argue that at least one of the sums must be large. In the first summation, since $\theta_t^\alpha < -R$, we must have $\mathrm{cov}_t^\alpha = 0$, so $\mathrm{cov}_t^\alpha - \alpha = -\alpha$ is negative; thus each summand is positive. In the second summation, since $\theta_t^\alpha > R$, we must have $\mathrm{cov}_t^\alpha = 1$, so $\mathrm{cov}_t^\alpha - \alpha = 1 - \alpha$ is positive; thus each of these summands is also positive.

To see that at least one of the sums must be large, observe that since $h \geq Rm^{3/2}/d_{\mathcal{A}}$ by assumption, we have $|\theta_t^{\alpha^*}| > h/\sqrt{m} \geq Rm/d_{\mathcal{A}} \geq R$. Thus we know that α^* must appear in the indices of one of the first two summations. If α^* appears in the first summation, this means $\theta_t^{\alpha^*} < -h/\sqrt{m}$, so the first summation can be lower bounded by $h\alpha^*/\sqrt{m}$. If α^* appears in the second summation, the second summation can be similarly lower bounded by $h(1-\alpha^*)/\sqrt{m}$. Combining, we conclude that the first two sums in (31) can be lower bounded as

$$\sum_{\alpha:\theta_t^\alpha<-R} \theta_t^\alpha(\mathrm{cov}_t^\alpha - \alpha) + \sum_{\alpha:\theta_t^\alpha>R} \theta_t^\alpha(\mathrm{cov}_t^\alpha - \alpha) \geq \frac{h\min(\alpha^*, 1-\alpha^*)}{\sqrt{m}} \geq \frac{hd_{\mathcal{A}}}{\sqrt{m}}.$$

The third sum in (31) is lower bounded by $-Rm$, since $\mathrm{cov}_t^\alpha - \alpha \in [-1, 1]$. Plugging this all back into (31), we get

$$\langle \theta_t, g_t(\theta_t)\rangle \geq \frac{hd_{\mathcal{A}}}{\sqrt{m}} - Rm. \tag{32}$$

Note that the right-hand side is nonnegative for any $h \geq Rm^{3/2}/d_{\mathcal{A}}$.

B.2.2 Proof of Lemma 2

We will show that for any $\theta_t \in C_t = \mathcal{K} - b_t$, there exists $\delta > 0$ such that $\theta_t - \varepsilon g_t(\theta_t) \in C_t$, for all $\varepsilon \leq \delta$. That is, we will show that if we took a small enough step in the direction of the negative gradient in (11) starting from θ_t, then the quantiles would remain uncrossed. We do so by first arguing that we do not have to worry about crossings between quantiles on the same side of y_t, then arguing that the quantiles which sandwich y_t must be separated by a positive distance, allowing us to maintain proper ordering for small enough δ.

For $\varepsilon > 0$, let $\omega = \theta_t - \varepsilon g_t(\theta_t)$, with elements $\omega^\alpha = \theta_t^\alpha - \varepsilon(\mathrm{cov}_t^\alpha - \alpha)$, $\alpha \in \mathcal{A}$. We want to show that for small enough ε, we have $\omega \in C_t$. In other words, we must verify $\omega^{\alpha_i} + b_t^{\alpha_i} \leq \omega^{\alpha_{i+1}} + b_t^{\alpha_{i+1}}$ for $i = 1, \ldots, m-1$. First we show that for any pair $\alpha < \beta$, if $\mathrm{cov}_t^\alpha = \mathrm{cov}_t^\beta$, then $\omega^\alpha + b_t^\alpha \leq \omega^\beta + b_t^\beta$ for any $\varepsilon > 0$. To see this, observe

$$\begin{aligned}\omega^\beta + b_t^\beta - (\omega^\alpha + b_t^\alpha) &= \theta_t^\beta - \varepsilon(\mathrm{cov}_t^\beta - \beta) + b_t^\beta - [\theta_t^\alpha - \varepsilon(\mathrm{cov}_t^\alpha - \alpha) + b_t^\alpha] \\ &= \theta_t^\beta + b_t^\beta - (\theta_t^\alpha + b_t^\alpha) - \varepsilon(\mathrm{cov}_t^\beta - \mathrm{cov}_t^\alpha - \beta + \alpha) \\ &\geq \theta_t^\beta + b_t^\beta - (\theta_t^\alpha + b_t^\alpha) \\ &\geq 0,\end{aligned}$$

where the third line uses $\text{cov}_t^\alpha = \text{cov}_t^\beta$ and $\beta > \alpha$, and the fourth uses $\theta_t \in C_t$. Because

$$\theta_t^{\alpha_1} + b_t^{\alpha_1} \leq \theta_t^{\alpha_2} + b_t^{\alpha_2} \leq \cdots \leq \theta_t^{\alpha_m} + b_t^{\alpha_m},$$

we know that

$$0 \leq \text{cov}_t^{\alpha_1} \leq \text{cov}_t^{\alpha_2} \leq \cdots \leq \text{cov}_t^{\alpha_m} \leq 1.$$

Thus, there exists $k \in \{-1, 0, \ldots, m\}$ such that $\text{cov}_t^{\alpha_i} = 0$ for all $i \leq k$ and $\text{cov}_t^{\alpha_i} = 1$ for all $i > k$. For $i < k$ and $i > k$, we have $\omega^{\alpha_i} + b_t^{\alpha_i} \leq \omega^{\alpha_{i+1}} + b_t^{\alpha_{i+1}}$ for any $\varepsilon > 0$ by the fact above. The only case that remains to check is $i = k$. If $k = -1$ or $k = m$, then this means cov_t^α is the same for all quantile levels, so we are done, by the fact proven above. Now consider $0 \leq k \leq m-1$. Since $\text{cov}_t^{\alpha_k} = 0$ and $\text{cov}_t^{\alpha_{k+1}} = 1$, this implies

$$y_t \in (\theta_t^{\alpha_k} + b_t^{\alpha_k}, \theta_t^{\alpha_{k+1}} + b_t^{\alpha_{k+1}}],$$

which implies $(\theta_t^{\alpha_{k+1}} + b_t^{\alpha_{k+1}}) - (\theta_t^{\alpha_k} + b_t^{\alpha_k}) > 0$. Informally, since $\theta_t^{\alpha_k}$ and $\theta_t^{\alpha_{k+1}}$ are separated by a positive amount, we can increase $\theta_t^{\alpha_k}$ by a little and decrease $\theta_t^{\alpha_{k+1}}$ by a little and still maintain the ordering. Formally, setting $\delta = [(\theta_t^{\alpha_{k+1}} + b_t^{\alpha_{k+1}}) - (\theta_t^{\alpha_k} + b_t^{\alpha_k})]/2$, we see that for any $\varepsilon \leq \delta$,

$$\begin{aligned}
\omega^{\alpha_{k+1}} + b_t^{\alpha_{k+1}} - (\omega^{\alpha_k} + b_t^{\alpha_k}) &= \theta_t^{\alpha_{k+1}} - \varepsilon(\text{cov}_t^{\alpha_{k+1}} - \alpha_{k+1}) + b_t^{\alpha_{k+1}} - [\theta_t^{\alpha_k} - \varepsilon(\text{cov}_t^{\alpha_k} - \alpha_k) + b_t^{\alpha_k}] \\
&\geq \theta_t^{\alpha_{k+1}} + b_t^{\alpha_{k+1}} - (\theta_t^{\alpha_k} + b_t^{\alpha_k}) - 2\varepsilon \\
&\geq 0,
\end{aligned}$$

where the second line is due to $|\text{cov}_t^\alpha - \alpha| \leq 1$ for any $\alpha \in [0, 1]$, and the third line is due to the choice of ε.

B.3 Regret of lazy gradient descent with delay

To build up towards a proof of Theorem 4, we first derive a general bound on the regret of lazy gradient descent with delay, in a setting with time-varying constraint sets $C_t \subseteq \mathbb{R}^m$, $t = 1, 2, \ldots$. The time-varying constraint sets are what makes this problem unusual, and to our knowledge, standard regret bounds do not apply in this setting. While it might be possible to adapt more general results for lazy gradient updates (or follow the regularized leader) under adaptive regularization, e.g., as surveyed in McMahan (2017), we provide a relatively simple and self-contained analysis below, which leverages inward flow.

Theorem 5. *Assume that for each t, the loss function ℓ_t is L-Lipschitz and convex, the set C_t is closed and convex, and the pair (ℓ_t, C_t) satisfies inward flow. Then, for all $T \geq 1$ and $u \in \mathbb{R}^m$, the lazy gradient descent iterates produced by (18) and (17) satisfy*

$$\frac{1}{T}\sum_{t=1}^T \ell_t(\theta_t) - \frac{1}{T}\sum_{t=1}^T \ell_t(u) \leq \frac{\|\tilde{\theta}_1 - u\|_2^2}{2\eta T} + \frac{\eta(2D+1)L^2}{2}.$$

Proof. By convexity, $\ell_t(u) \geq \ell_t(\theta_t) + \langle g_t(\theta_t), u - \theta_t \rangle$. Rearranging and summing over t gives

$$\sum_{t=1}^T (\ell_t(\theta_t) - \ell_t(u)) \leq \sum_{t=1}^T \langle g_t(\theta_t), \theta_t - u \rangle. \tag{33}$$

By the representation (27), note that we can write

$$\tilde{\theta}_1 - \eta \sum_{s=1}^{t-D-1} g_s(\theta_s) = \theta_t + v_t$$

for some $v_t \in N_{C_t}(\theta_t)$. (Recall our convention that we set $g_t(\theta_t) = 0$ for $t \leq 0$.) In other words,

$$\theta_t = \tilde{\theta}_1 - \eta \sum_{s=1}^{t-D-1} g_s(\theta_s) - v_t,$$

and therefore we have

$$\begin{aligned}
\sum_{t=1}^{T} \langle g_t(\theta_t), \theta_t \rangle &= \sum_{t=1}^{T} \langle g_t(\theta_t), \tilde{\theta}_1 \rangle - \eta \sum_{t=1}^{T} \Big\langle g_t(\theta_t), \sum_{s=1}^{t-D-1} g_s(\theta_s) \Big\rangle - \sum_{t=1}^{T} \langle g_t(\theta_t), v_t \rangle \\
&\leq \sum_{t=1}^{T} \langle g_t(\theta_t), \tilde{\theta}_1 \rangle - \eta \sum_{t=1}^{T} \Big\langle g_t(\theta_t), \sum_{s=1}^{t-D-1} g_s(\theta_s) \Big\rangle,
\end{aligned}$$

where the second line holds because each summand in the third sum satisfies $\langle g_t(\theta_t), v_t \rangle \geq 0$: if $\theta_t \in \mathrm{int}(C_t)$, then $v_t = 0$, otherwise if $\theta_t \in \mathrm{bd}(C_t)$, then this inner product is nonnegative by part (i) of Lemma 4 as a consequence of inward flow. Plugging this into (33) gives

$$\sum_{t=1}^{T} (\ell_t(\theta_t) - \ell_t(u)) \leq \sum_{t=1}^{T} \langle g_t(\theta_t), \tilde{\theta}_1 - u \rangle - \eta \sum_{t=1}^{T} \Big\langle g_t(\theta_t), \sum_{s=1}^{t-D-1} g_s(\theta_s) \Big\rangle, \tag{34}$$

and from here on, we follow standard arguments for online gradient descent (or follow the regularized leader, more generally). Beginning with the second term in (34), observe

$$\begin{aligned}
\Big\langle g_t(\theta_t), \sum_{s=1}^{t-D-1} g_s(\theta_s) \Big\rangle &= \Big\langle g_t(\theta_t), \sum_{s=1}^{t-1} g_s(\theta_s) \Big\rangle - \Big\langle g_t(\theta_t), \sum_{s=t-D}^{t-1} g_s(\theta_s) \Big\rangle \\
&\geq \Big\langle g_t(\theta_t), \sum_{s=1}^{t-1} g_s(\theta_s) \Big\rangle - DL^2,
\end{aligned} \tag{35}$$

where the second line uses Lipschitzness. For the first term above, note that

$$\Big\langle g_t(\theta_t), \sum_{s=1}^{t-1} g_s(\theta_s) \Big\rangle = \frac{1}{2} \left(\Big\| \sum_{s=1}^{t} g_s(\theta_s) \Big\|_2^2 - \Big\| \sum_{s=1}^{t-1} g_s(\theta_s) \Big\|_2^2 - \|g_t(\theta_t)\|_2^2 \right).$$

Summing over $t = 1, \ldots, T$, the right-hand side telescopes, yielding

$$\sum_{t=1}^{T} \Big\langle g_t(\theta_t), \sum_{s<t} g_s(\theta_s) \Big\rangle = \frac{1}{2} \Big\| \sum_{t=1}^{T} g_t(\theta_t) \Big\|_2^2 - \frac{1}{2} \sum_{t=1}^{T} \|g_t(\theta_t)\|_2^2.$$

Substituting back into (35), we get

$$\begin{aligned}
\sum_{t=1}^{T} \Big\langle g_t(\theta_t), \sum_{s=1}^{t-D-1} g_s(\theta_s) \Big\rangle &\geq \frac{1}{2} \Big\| \sum_{t=1}^{T} g_t(\theta_t) \Big\|_2^2 - \frac{1}{2} \sum_{t=1}^{T} \|g_t(\theta_t)\|_2^2 - DL^2T \\
&\geq \frac{1}{2} \Big\| \sum_{t=1}^{T} g_t(\theta_t) \Big\|_2^2 - \frac{1}{2}(2D+1)L^2T,
\end{aligned}$$

where the second line again uses Lipschitzness, and from (34) we then have

$$\sum_{t=1}^{T} (\ell_t(\theta_t) - \ell_t(u)) \leq \sum_{t=1}^{T} \langle g_t(\theta_t), \tilde{\theta}_1 - u \rangle - \frac{\eta}{2} \Big\| \sum_{t=1}^{T} g_t(\theta_t) \Big\|_2^2 + \frac{\eta}{2}(2D+1)L^2T. \tag{36}$$

Now we bound the first two terms in (36). Observe

$$0 \le \left\| \tilde{\theta}_1 - u - \eta \sum_{t=1}^{T} g_t(\theta_t) \right\|_2^2 = \|\tilde{\theta}_1 - u\|_2^2 - 2\eta \left\langle \tilde{\theta}_1 - u, \sum_{t=1}^{T} g_t(\theta_t) \right\rangle + \eta^2 \left\| \sum_{t=1}^{T} g_t(\theta_t) \right\|_2^2,$$

which implies

$$\sum_{t=1}^{T} \langle g_t(\theta_t), \tilde{\theta}_1 - u \rangle \le \frac{1}{2\eta} \|\tilde{\theta}_1 - u\|_2^2 + \frac{\eta}{2} \left\| \sum_{t=1}^{T} g_t(\theta_t) \right\|_2^2.$$

Plugging this into (36) yields the desired result. □

B.4 Regret theory for MultiQT

In this section, we prove the regret bound for MultiQT, which follows easily by combining our above result on the regret of lazy mirror descent with a lemma characterizing the optimal comparator.

B.4.1 Proof of Theorem 4

This is a direct consequence of Theorem 5, specialized to the MultiQT setting. By this result, for any θ (which was written as u in the previous theorem),

$$\begin{aligned} \frac{1}{T} \sum_{t=1}^{T} \ell_t(\theta_t) - \frac{1}{T} \sum_{t=1}^{T} \ell_t(\theta) &\le \frac{\|\tilde{\theta}_1 - \theta\|_2^2}{2\eta T} + \frac{\eta(2D+1)L^2}{2} \\ &\le \frac{\|\tilde{\theta}_1\|_2^2}{\eta T} + \frac{\|\theta\|_2^2}{\eta T} + \frac{\eta(2D+1)L^2}{2}, \end{aligned}$$

where the second line uses $\|a+b\|_2^2 \le 2\|a\|_2^2 + 2\|b\|_2^2$. Now we plug in an optimal point θ^*, which is defined to minimize the aggregate quantile loss, and we use the optimal comparator lemma below, which says that $\|\theta^*\|_2^2 \le R^2 m$. This completes the proof.

B.4.2 Optimal comparator lemma

We state and prove a result used above regarding the ℓ_2 norm of the optimal comparator in the MultiQT setting.

Lemma 5. *Let θ^* be an optimal fixed offset in hindsight, according to the MultiQT loss — that is, θ^* minimizes $\sum_{t=1}^{T} \ell_t(\theta)$ over all $\theta \in \mathbb{R}^m$, where $\ell_t(\theta) = \rho_{\mathcal{A}}(b_t + \theta, y_t)$. Then, under the conditions of Theorem 4, we have $\|\theta^*\|_2^2 \le R^2 m$.*

Proof. In brief, θ^* is the vector of empirical quantiles of the residuals, which is guaranteed to lie in between the extremes of these residuals, implying the claimed result after a comparison inequality between ℓ_∞ and ℓ_2 norms. In more detail, recall that in (8) we defined the quantile loss ρ_α for a single level α. From the definition, it is immediate that

$$\rho_\alpha(b_t^\alpha + \theta^\alpha, y_t) = \rho_\alpha(\theta_t^\alpha, y_t - b_t^\alpha),$$

and therefore the MultiQT loss ℓ_t defined in (10) can be rewritten as

$$\ell_t(\theta) = \sum_{\alpha \in \mathcal{A}} \rho_\alpha(\theta_t^\alpha, y_t - b_t^\alpha).$$

Summing this over $t = 1, \ldots, T$, we get

$$\sum_{t=1}^{T} \sum_{\alpha \in \mathcal{A}} \rho_\alpha(\theta_t^\alpha, y_t - b_t^\alpha) = \sum_{\alpha \in \mathcal{A}} \sum_{t=1}^{T} \rho_\alpha(\theta_t^\alpha, y_t - b_t^\alpha),$$

of which θ^* is the minimizer over all $\theta \in \mathbb{R}^m$. This minimization decouples into separate minimizations per quantile level. For each level α, by a standard result, minimizers of the loss

$$\sum_{t=1}^{T} \rho_\alpha(\theta_t^\alpha, y_t - b_t^\alpha)$$

are empirical α-level quantiles of the given data, here the residuals $y_t - b_t^\alpha$, $t = 1, \ldots, T$. In general, this will not be unique, but any such minimizer will lie in between the maximum and minimum values of the data, which are bounded by $-R$ and R by assumption. Hence, we know that $\|\theta^*\|_\infty \leq R$, and therefore $\|\theta^*\|_2 \leq R\sqrt{m}$ by a standard comparison inequality between ℓ_∞ and ℓ_2 norms. □

C Miscellaneous negative results

In this section, we prove the negative results stated in the paper.

C.1 Proof of Proposition 2

We prove this by constructing a counterexample, which is valid both when G is the sorting operator and the isotonic projection operator. Take $b_t^\alpha = 0$ for all α and t. This implies $q_t = \theta_t$, so below we will reference θ_t directly. For simplicity, we consider only two quantile levels $\mathcal{A} = \{\alpha, \beta\}$, where $\alpha = 0.5$ and $\beta = 0.75$. Recall that θ_t^α is the α-level QT forecast at time t, and let $\hat{\theta}_t^\alpha$ denote the α-level forecast after applying the map G. We initialize $\theta_1^\alpha = \theta_1^\beta = 0$. We now define a sequence y_t with crossing events at a nonvanishing fraction of time steps, resulting in the incorrect long-run coverage after applying G. Consider the following sequence of outcomes, visualized in Figure 11:

- y_1 lands above both forecasts, so both forecasts increase and become separated by a positive gap;
- y_2 lands in this gap, so the α-level forecast decreases and the β-level forecast increases, and the quantiles are now crossed;
- y_3 lands in between the two crossed quantiles;
- y_4 through y_8 are a sequence of values that cause the forecasts to reset to the starting point of zero at time $t = 9$, at which point we repeat the entire subsequence ad infinitum.

Of the eight time steps in each subsequence, θ_t^α covers y_t four times, yielding the desired coverage of 0.5, and θ_t^β covers y_t six times out of eight, yielding the desired coverage of 0.75.

When G is the sorting operator, the crossing means the coverage events at the third time step are swapped, causing the sorted quantiles $\hat{\theta}_t^\alpha$ and $\hat{\theta}_t^\beta$ to yield coverages of 3/8 and 7/8, respectively.

When G is the isotonic regression operator, the crossing at the third time step similarly causes a problem. By the pool adjacent violators algorithm (PAVA) (Barlow et al., 1972), we know that isotonic regression maps any pair of crossed quantiles to the same value. Thus, rather than swapping the coverage events at $t = 3$, the use of isotonic regression causes one of the coverage events to flip. Let $\hat{\theta}_3^*$ denote the common

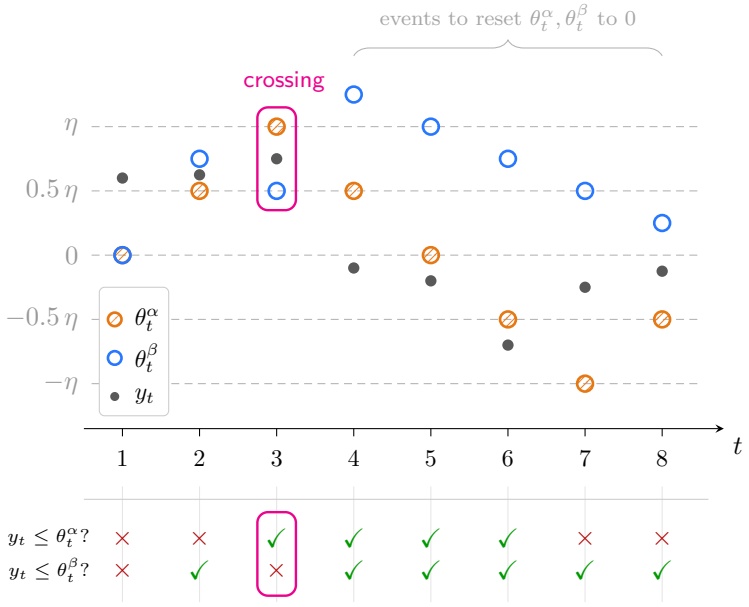


Figure 11: An example where post hoc ordering the QT iterates (via either sorting or isotonic regression) fails to achieve the correct coverage with two quantile levels, $\alpha = 0.5$ and $\beta = 0.75$. The sequence y_t relative to θ_t^α and θ_t^β is (1) above both, (2) in between, (3) in between the crossed quantiles, (4-6) below both, (7-8) in between both. Both forecasts are initialized to zero at time $t = 1$ and return to zero at time $t = 9$, at which point the sequence of y_t is repeated. Success and failure of coverage is marked with ✓ and ×, respectively. Averaging across each row, we see that θ_t^α achieves a coverage of 0.5 and θ_t^β a coverage of 0.75, as desired. However, after applying post hoc ordering, the coverage events at $t = 3$ are modified: for sorting, the coverage events are swapped with each other, and for isotonic regression, either both coverage events become successes or both become failures. In all cases, the ordered iterates fail to achieve the correct coverage for at least one quantile level.

value that θ_3^α and θ_3^β are mapped to by isotonic regression (their average). Now, if $y_3 \le \hat{\theta}_3^*$, then the coverage indicators for the ordered quantiles at $t = 3$ will both be one; conversely, if $y_3 > \hat{\theta}_3^*$, then both indicators will both be zero. In either case, coverage will fail to be obtained by the ordered quantiles at one of the levels (the one whose indicators flipped after applying isotonic regression).

C.2 Proof of Proposition 4

We construct a counterexample where projected gradient descent fails to yield coverage. As in the proof of Proposition 2, let $b_t^\alpha = 0$ for all α and t, and $\mathcal{A} = \{\alpha, \beta\}$, where now $\alpha < \beta$ and $\alpha + \beta = 0.5$. Initialize $\theta_1^\alpha = \theta_1^\beta = q$ for some $q \in \mathbb{R}$. Suppose we observe $y_1 > q$, so $\tilde{\theta}_2^\alpha = q + \eta\alpha$ and $\tilde{\theta}_2^\beta = q + \eta\beta$. Since $\alpha < \beta$, the quantiles are ordered and therefore we have $\theta_2^\alpha = \tilde{\theta}_2^\alpha$ and $\theta_2^\beta = \tilde{\theta}_2^\beta$.

Now suppose we observe $y_2 \in (\theta_2^\alpha, \theta_2^\beta]$, so we updates the hidden iterates to

$$\tilde{\theta}_3^\alpha = q + 2\eta\alpha,$$
$$\tilde{\theta}_3^\beta = q + \eta\beta - \eta(1 - \beta) = q + \eta(2\beta - 1).$$

Since $\beta = 0.5 - \alpha$, we have $2\beta - 1 = -2\alpha$, so a crossing has occurred: $\tilde{\theta}_3^\alpha > \tilde{\theta}_3^\beta$. By the pool adjacent violators

algorithm (PAVA) (Barlow et al., 1972), we know that isotonic regression will map these two values to their average:

$$\theta_3^\alpha = \theta_3^\beta = \frac{\tilde{\theta}_3^\alpha + \tilde{\theta}_3^\beta}{2} = q + \frac{\eta(2(\alpha+\beta)-1)}{2} = q,$$

where the last equality uses $\alpha+\beta = 0.5$. This puts us back to the starting point from $t = 1$; we can therefore repeat this process, so that the α-level forecasts achieve a coverage of 0 and the β-level forecasts achieve a coverage of 0.5.

C.3 MultiQT with sorting

The next result highlights the importance of using isotonic projection to enforce the ordering constraints in MultiQT, as it shows that replacing this with sorting does not achieve calibration in general.

Proposition 8. *Let $q_t, t = 1, 2, \ldots$ be forecasts obtained by running Procedure 1 but where the projection step in (5) is replaced with sorting—that is, $q_t \in \mathbb{R}^m$ is set equal to the vector which results from sorting the entries of $b_t + \tilde{\theta}_t \in \mathbb{R}^m$. Then, there exists a set of levels $\mathcal{A}$ and sequence of target values and base forecasts (y_t, b_t) with bounded errors (i.e., $|y_t - b_t^\alpha|$ is bounded for all α and t) such that for any learning rate $\eta > 0$ the forecasts fail to achieve calibration : $\lim_{T\to\infty} \frac{1}{T}\sum_{t=1}^T \mathbb{1}\{y_t \le q_t^\alpha\} \neq \alpha$ for some $\alpha \in \mathcal{A}$.*

Proof. As in previous counterexamples, we set $b_t^\alpha = 0$ for all α and t, thus $q_t = \theta_t$ for all t, and $\mathcal{A} = \{\alpha, \beta\}$, where $0 < \alpha < \beta < 1$ and $\alpha = 1 - \beta$. Initialize $\tilde{\theta}_1^\alpha < \tilde{\theta}_1^\beta$, such that $\tilde{\theta}_1^\beta - \tilde{\theta}_1^\alpha < 2\eta\alpha$. Since these are ordered, we have $\theta_1^\alpha = \tilde{\theta}_1^\alpha$ and $\theta_1^\beta = \tilde{\theta}_1^\beta$.

Suppose we observe $y_1 \in (\theta_1^\alpha, \theta_1^\beta]$, so the hidden iterate updates are

$$\tilde{\theta}_2^\alpha = \tilde{\theta}_1^\alpha + \eta\alpha,$$
$$\tilde{\theta}_2^\beta = \tilde{\theta}_1^\beta - \eta(1-\beta).$$

These are crossed, as $\tilde{\theta}_2^\beta - \tilde{\theta}_2^\alpha = \tilde{\theta}_1^\beta - \tilde{\theta}_1^\alpha - \eta(1-\beta+\alpha) = \tilde{\theta}_1^\beta - \tilde{\theta}_1^\alpha - 2\eta\alpha < 0$, by assumption. Sorting yields the played updates: $\theta_2^\alpha = \tilde{\theta}_2^\beta$ and $\theta_2^\beta = \tilde{\theta}_2^\alpha$, and the key realization for this counterexample is that the played iterates now have a *positive* gap (instead of zero gap, as with isotonic projection), which can be exploited to continue driving them farther away from each other.

In particular, suppose $y_2 \in (\theta_2^\alpha, \theta_2^\beta]$, so the hidden iterate updates are

$$\tilde{\theta}_3^\alpha = \tilde{\theta}_2^\alpha + \eta\alpha,$$
$$\tilde{\theta}_3^\beta = \tilde{\theta}_2^\beta - \eta(1-\beta).$$

Since $\tilde{\theta}_2^\alpha$ and $\tilde{\theta}_2^\beta$ are already crossed, this update keeps them crossed (and increases the gap in between them). Sorting yields played iterates $\theta_3^\alpha = \tilde{\theta}_3^\beta$ and $\theta_3^\beta = \tilde{\theta}_3^\alpha$. This can be repeated ad infinitum, causing $\tilde{\theta}_t^\alpha \to \infty$ and $\tilde{\theta}_t^\beta \to -\infty$ as $t \to \infty$. Hence, the β-level coverage goes to 1 and α-level coverage goes to 0. □

C.4 MultiQT with positively separated quantiles

The next result highlights the importance of not only projection, but specifically projection to the (shifted) isotonic cone; modifying the constraint set to induce quantiles separated by $\varepsilon > 0$ fails to achieve calibration in general.

Proposition 9. *Consider the set defined in (25), which can be equivalently written as $C_t^\varepsilon = \mathcal{K}^\varepsilon - b_t$, where*

$$\mathcal{K}^\varepsilon = \left\{ x \in \mathbb{R}^m : x_i + \varepsilon \le x_{i+1},\ i = 1, 2, \ldots, m-1 \right\}$$

for $\varepsilon > 0$. Let $q_t, t = 1, 2, \ldots$ be forecasts obtained by running Procedure 1 but where the projection step in (5) is replaced with $q_t = \Pi_{\mathcal{K}^\varepsilon}(b_t + \tilde{\theta}_t)$. Then, for any $\varepsilon > 0$, there exists a set of levels $\mathcal{A}$ and sequence of target values and base forecasts (y_t, b_t) with bounded errors (i.e., $|y_t - b_t^\alpha|$ is bounded for all α and t) such that for any learning rate $\eta > 0$ the forecasts fail to achieve calibration : $\lim_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T} \mathbb{1}\{y_t \le q_t^\alpha\} \neq \alpha$ for some $\alpha \in \mathcal{A}$.

Proof. We adapt the construction from the proof of Proposition 8. Consider the same initialization and initial target y_1. Under projection onto $\mathcal{K}^\varepsilon$, the played updates satisfy $\theta_2^\beta - \theta_2^\alpha \ge \varepsilon$, creating a strictly positive gap. By choosing $y_2 \in (\theta_2^\alpha, \theta_2^\beta]$, we can continue driving the played iterates away from one another. This results in the same limiting behavior as in the previous construction, where coverage at the upper level converges to 1 and coverage at the lower level converges to 0. □

D Fast rates for constrained gradient equilibrium

We derive fast rates for constrained gradient equilibrium and, consequently, calibration without crossings by refining the analysis to leverage a positive curvature assumption. We note that these results could be extended to the setting of delayed feedback, but for simplicity, we state and prove all results in the no-delay setting ($D = 0$).

D.1 Proof of Proposition 7

We follow the general proof structure from Proposition 5 of Angelopoulos et al. (2025). For convenience, let us redefine $h_t = \max\{\|\tilde{\theta}_1\|_2, h_t\}$ and let $h_0 = \|\tilde{\theta}_1\|_2$. We will use induction to show $\|\tilde{\theta}_{T+1}\|_2 \le h_T + B + \eta L$ for all $T \ge 0$. The base case for $T = 0$ holds trivially. For the inductive step, assume the inequality holds up through T. We split into two cases. First, if $\|\tilde{\theta}_T\|_2 \le h_T + B$, then by the triangle inequality we have

$$\begin{aligned} \|\tilde{\theta}_{T+1}\|_2 &\le \|\tilde{\theta}_T\|_2 + \eta \|g_T(\theta_T)\|_2 \\ &\le h_T + B + \eta L, \end{aligned}$$

where the second inequality invokes Lipschitzness of the loss. Second, if $\|\tilde{\theta}_T\|_2 > h_T + B$, then

$$\begin{aligned} \|\tilde{\theta}_{T+1}\|_2^2 &= \|\tilde{\theta}_T\|_2^2 + \eta^2 \|g_T(\theta_T)\|_2^2 - 2\eta \langle \tilde{\theta}_T, g_T(\theta_T) \rangle \\ &\le \|\tilde{\theta}_T\|_2^2 + \eta^2 L^2 - 2\eta \langle \tilde{\theta}_T, g_T(\theta_T) \rangle \\ &\le \|\tilde{\theta}_T\|_2^2 + \eta^2 L^2 - 2\eta \phi_T(\theta_T) \\ &\le \|\tilde{\theta}_T\|_2^2 \\ &\le (h_T + B + \eta L)^2 \\ &\le (h_{T+1} + B + \eta L)^2 \end{aligned}$$

where second line applies Lipschitzness, the third is discussed below, the fourth uses the assumed curvature condition on $\phi_T(\theta_T)$, the fifth applies the inductive hypothesis, and the sixth uses the increasing property of h_t. Taking a square root would conclude the inductive step.

It remains to verify the third line above, which uses $\langle \tilde{\theta}_T, g_T(\theta_T) \rangle \geq \phi_T(\theta_T)$. This is due to restorativity, inward flow, and the bounded distance assumption. In particular, note that by the triangle inequality

$$\|\theta_T\|_2 > \|\tilde{\theta}_T\|_2 - \|\theta_T - \tilde{\theta}_T\|_2 \geq h + B - B = h.$$

Thus, by restorativity of ℓ, we have $\langle \theta_T, g_T(\theta_T) \rangle \geq \phi_T(\theta_T)$. By inward flow, we can then apply part (ii) of Lemma 4 to get $\langle \tilde{\theta}_T, g_T(\theta_T) \rangle \geq \phi_T(\theta_T)$ as desired. This completes the proof of the iterate bound.

For the average gradient bound, we simplify the iterate bound using $\max\{a, b\} \leq a + b$, and then apply (28) with $D = 0$.

D.2 Proof of Lemma 3

Our proof has two main steps. First, we will show that if the base forecaster is a point forecaster, then the entries of $\tilde{\theta}_t$ cannot get too crossed: specifically, $\tilde{\theta}_t^{\alpha_i} - \tilde{\theta}_t^{\alpha_{i+1}} \leq \eta$ for all $i = 1, 2, \ldots, m-1$. We then bound the projection distance $\|\tilde{\theta}_t - \theta_t\|_2 = \|\tilde{\theta}_t - \Pi_{\mathcal{K}-b_t}(\tilde{\theta}_t)\|_2$ for any $\tilde{\theta}_t$ satisfying this crossing bound to obtain the desired result.

Before beginning with the first step, we make the following important observation about the MultiQT iterates when the base forecasts are point forecasts: if $b_t^\alpha = \mu_t$ for all α, then the played iterate is simply the result of running isotonic regression on the hidden iterate, i.e.,

$$\theta_t = \Pi_{\mathcal{K}}(\tilde{\theta}_t).$$

To see why, recall that by definition $\theta_t = \Pi_{\mathcal{K}-b_t}(\tilde{\theta}_t)$, but for $b_t = \mu_t \mathbf{1}$, where $\mathbf{1} \in \mathbb{R}^m$ denotes the vector of all ones, the isotonic cone is shift-invariant: $\mathcal{K} - \mu_t \mathbf{1} = \mathcal{K}$.

We now show that $\tilde{\theta}_t^{\alpha_i} - \tilde{\theta}_t^{\alpha_{i+1}} \leq \eta$ for all $i = 1, 2, \ldots, m-1$ and all times t, by induction on t. The base case holds by assumption: $\tilde{\theta}_1$ in the lemma is assumed to lie in $\mathcal{K}$. Now assume the statement holds through time t. Define $\Delta_t^i = \tilde{\theta}_t^{\alpha_{i+1}} - \tilde{\theta}_t^{\alpha_i}$, where $\Delta_t^i < 0$ means a crossing has occurred, and $\Delta_t^i \geq 0$ means entries i and $i+1$ of $\tilde{\theta}_t$ are ordered. Fix any i. We break our analysis into two cases.

- *Case 1*: $\Delta_t^i < 0$ (the entries are crossed at time t). In this case, isotonic regression will pool entries i and $i+1$ so that $q_t^{\alpha_i} = q_t^{\alpha_{i+1}}$ (Barlow et al., 1972), which implies $\mathrm{cov}_t^{\alpha_i} = \mathrm{cov}_t^{\alpha_{i+1}}$. Thus,

  $$\Delta_{t+1}^i = \Delta_t^i - \eta[(\mathrm{cov}_t^{\alpha_{i+1}} - \alpha_{i+1}) - (\mathrm{cov}_t^{\alpha_i} - \alpha_i)] = \Delta_t^i + \eta(\alpha_{i+1} - \alpha_i) \geq \Delta_t^i \geq -\eta,$$

  where the last inequality follows from the inductive hypothesis.

- *Case 2*: $\Delta_t^i \geq 0$ (the entries are ordered at time t). In this case, $q_t^{\alpha_i} \leq q_t^{\alpha_{i+1}}$, so $\mathrm{cov}_t^{\alpha_{i+1}} \geq \mathrm{cov}_t^{\alpha_i}$, and

  $$\Delta_{t+1}^i = \Delta_t^i - \eta[(\mathrm{cov}_t^{\alpha_{i+1}} - \mathrm{cov}_t^{\alpha_i}) - (\alpha_{i+1} - \alpha_i)] \geq \Delta_t^i - \eta(1 - (\alpha_{i+1} - \alpha_i)).$$

  Since $\Delta_t^i \geq 0$ and $(\alpha_{i+1} - \alpha_i) > 0$, it follows that $\Delta_{t+1}^i > -\eta$.

In both cases, $\Delta_{t+1}^i \geq -\eta$, which establishes $\tilde{\theta}_{t+1}^{\alpha_i} - \tilde{\theta}_{t+1}^{\alpha_{i+1}} \leq \eta$, completing the inductive proof.

Now fix any $\tilde{\theta}_t$ satisfying $\tilde{\theta}_t^{\alpha_i} - \tilde{\theta}_t^{\alpha_{i+1}} \leq \eta$ for all $i = 1, 2, \ldots, m-1$. Consider constructing the ordered vector $\bar{\theta}_t$ as follows: iterate through the indices of $\tilde{\theta}_t$ and, whenever we encounter an unordered entry, we set its value equal to that of the previous index. To make this more explicit:

- we set $\bar{\theta}_t^{\alpha_1} = \tilde{\theta}_t^{\alpha_1}$;

- for $i = 2, \ldots, m-1$, we set

$$\bar{\theta}_t^{\alpha_i} = \begin{cases} \bar{\theta}_t^{\alpha_{i-1}} & \text{if } \tilde{\theta}_t^{\alpha_i} < \bar{\theta}_t^{\alpha_{i-1}}, \\ \tilde{\theta}_t^{\alpha_i} & \text{otherwise.} \end{cases}$$

Since $\bar{\theta}_t \in \mathcal{K}$, we have

$$\|\tilde{\theta}_t - \Pi_{\mathcal{K}}(\tilde{\theta}_t)\|_2 \leq \|\tilde{\theta}_t - \bar{\theta}_t\|_2. \tag{37}$$

To bound the right-hand side, observe that $\tilde{\theta}_t^{\alpha_1} - \bar{\theta}_t^{\alpha_1} = 0$ and, for any $i \geq 2$,

$$\tilde{\theta}_t^{\alpha_i} - \bar{\theta}_t^{\alpha_i} = \begin{cases} \tilde{\theta}_t^{\alpha_i} - \bar{\theta}_t^{\alpha_{i-1}} & \text{if } \tilde{\theta}_t^{\alpha_i} < \bar{\theta}_t^{\alpha_{i-1}}, \\ 0 & \text{otherwise.} \end{cases}$$

As $\tilde{\theta}_t^{\alpha_i} \geq \tilde{\theta}_t^{\alpha_{i-1}} - \eta$, the above display implies

$$\tilde{\theta}_t^{\alpha_{i-1}} - \bar{\theta}_t^{\alpha_{i-1}} - \eta \leq \tilde{\theta}_t^{\alpha_i} - \bar{\theta}_t^{\alpha_i} \leq 0.$$

Therefore $|\tilde{\theta}_t^{\alpha_i} - \bar{\theta}_t^{\alpha_i}| \leq \eta(i-1)$, and

$$\|\tilde{\theta}_t - \bar{\theta}_t\|_2 \leq \sqrt{\sum_{i=1}^m (\eta(i-1))^2} = \eta \sqrt{\sum_{i=1}^m (i-1)^2} = \eta \sqrt{\frac{m(m-1)(2m-1)}{6}} \leq \frac{\eta m^{3/2}}{\sqrt{3}}.$$

Plugging this bound back into (37) completes the proof.

D.3 Proof of Corollary 1

If we inspect the conclusion (32) from the proof of Lemma 1 carefully, then we see what was shown here is actually stronger than the conclusion stated in the lemma. Rephrased, this proof showed that for any $c \geq 1$, the MultiQT loss ℓ_t is (h_c, ϕ_c)-restorative at all times t, for $h_c = cRm^{3/2}/d_{\mathcal{A}}$, and $\phi_c(\theta) = (c-1)Rm$. Thus to satisfy the positive curvature condition in Proposition 7, we require

$$(c-1)Rm \geq \eta L^2/2 \iff c \geq \eta L^2/(2Rm) + 1.$$

The smallest allowable value of c here is $c^* = \eta L^2/(2Rm) + 1$. Recalling that $L^2 = m$ for the MultiQT loss, this leads to the value $h^* = h_{c^*} = (\eta/2 + R)(m^{3/2}/d_{\mathcal{A}})$.

Note that we have shown that each MultiQT loss is $(h^*, \eta L^2/2)$-restorative. Since, additionally, the hidden and played iterates remain within an ℓ_2 distance of $B = \eta m^{3/2}/\sqrt{3}$ from each other by Lemma 3, we can apply Proposition 7, which yields the result.

E Additional experimental results

In this section, we provide supplementary empirical results that further illustrate the behavior of MultiQT across our forecasting datasets. First, we recompute the main experimental results using PIT entropy as an alternative calibration metric, in place of the ℓ_1 calibration error. We find that our conclusions remain qualitatively unchanged. Then, we present additional case studies (comprehensive calibration curves and forecast visualizations) for COVID-19 death forecasting and energy forecasting, which confirm that MultiQT consistently improves calibration.

E.1 Results using PIT entropy

PIT entropy (Gneiting et al., 2007; Rumack et al., 2022) is a calibration metric based on the entropy of the distribution of the probability integral transform (PIT) values, computed from forecasts and their corresponding targets. Specifically, given forecasts represented via cumulative distribution functions (CDFs) F_t, $t = 1, 2, 3, \ldots$, and associated target values y_t, $t = 1, 2, 3, \ldots$, the PIT values are defined as $F_t(y_t)$, $t = 1, 2, 3, \ldots$. The PIT entropy is then defined as the Shannon entropy of the empirical distribution of these PIT values.

To compute this entropy in practice, we divide the unit interval into $K = 10$ equal-width bins. Let $\hat{p}_k$ be the empirical frequency of PIT values for bin k. As our metric, we use the normalized Shannon entropy:

$$\hat{H} = -\frac{1}{\log K} \sum_{k=1}^{K} \hat{p}_k \log \hat{p}_k,$$

where the division by $\log K$ ensures that $\hat{H}$ lies in $[0, 1]$. Note that under perfect calibration, the PIT values should be distributed uniformly on $[0, 1]$, which has maximal entropy. Thus, a value of $\hat{H}$ near one indicates good calibration, while a value near zero indicates poor calibration.

In our setting, we have quantile forecasts q_t, $t = 1, 2, 3, \ldots$ and not CDFs. To construct CDFs from these forecasts (so that we can compute PIT entropy), we follow the procedure from Appendix A.1 of Buchweitz et al. (2025). This uses linear interpolation in between intermediate quantiles combined with exponential tails for values outside the extreme quantiles. Below we describe the procedure for constructing F_t from q_t.

- If there exists $i \leq m-1$ such that $y \in [q_t^{\alpha_i}, q_t^{\alpha_{i+1}}]$, then $F_t(y) = \alpha_i + \frac{y - q_t^{\alpha_i}}{q_t^{\alpha_{i+1}} - q_t^{\alpha_i}}(\alpha_{i+1} - \alpha_i)$.
  - Note that when ties occur (i.e., $q_t^{\alpha_{i+1}} = q_t^{\alpha_i}$), the interior slope on this segment is undefined, so when y equals a tied forecast value, we set $F_t(y)$ to be the largest quantile level in the tied block. Formally, $F_t(y) = \alpha^{i^*}$ where $i^* = \max\{i : q_t^{\alpha_i} = y\}$.
- If no such i exists (so $y < q_t^{\alpha_1}$ or $y > q_t^{\alpha_m}$), then we use exponential tails, chosen so that the density at the boundary matches that at the nearest interior segment.
  - For $y < q_t^{\alpha_1}$, we first find the smallest i such that $q_t^{\alpha_{i+1}} \neq q_t^{\alpha_i}$, and compute $\rho = \frac{\alpha_{i+1} - \alpha_i}{q_t^{\alpha_{i+1}} - q_t^{\alpha_i}}$. Define $\lambda = \frac{\rho}{\alpha_1}$. We then let $F_t(y) = \alpha_1 e^{\lambda(y - q_t^{\alpha_1})}$.
  - Similarly, for $y > q_t^{\alpha_m}$, we first find the largest i such that $q_t^{\alpha_{i+1}} \neq q_t^{\alpha_i}$, and compute $\rho = \frac{\alpha_{i+1} - \alpha_i}{q_t^{\alpha_{i+1}} - q_t^{\alpha_i}}$. Define $\lambda = \frac{\rho}{1 - \alpha_m}$. We then let $F_t(y) = 1 - (1 - \alpha_m) e^{-\lambda(y - q_t^{\alpha_m})}$.

After performing this procedure to translate a given sequence of quantile forecasts into CDFs, we compute the PIT values and PIT entropy as described above.

We now reproduce the main figures from Section 5 using PIT entropy in place of average calibration error, as used in the main text. Figure 12 is the analog of Figure 5, and Figure 13 the analog of Figure 7. We see qualitatively very similar trends, and MultiQT again results in strong improvements in calibration.

E.2 Additional COVID-19 forecasting results

To complement Figure 4 in the main paper, which shows actual versus desired coverage for one-week-ahead COVID-19 death forecasters before and after applying MultiQT, Figure 14 shows the same calibration plots for all forecasting horizons (one, two, three, and four weeks ahead). We observe that MultiQT consistently improves calibration across all forecasting horizons.

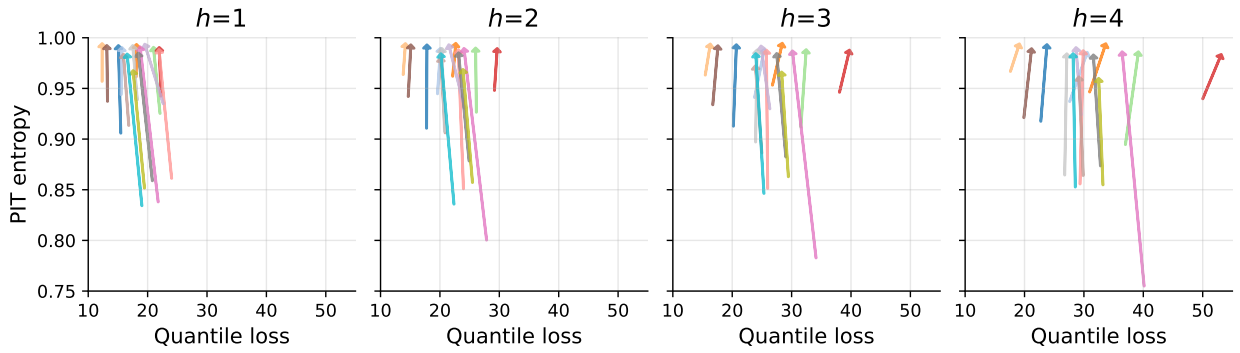
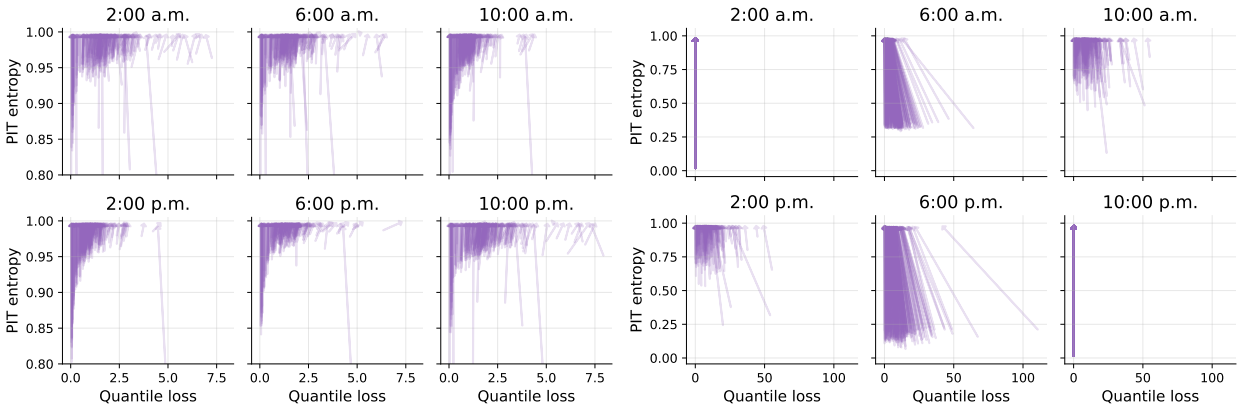


Figure 12: PIT entropy versus quantile loss on the COVID-19 death dataset, analogous to Figure 5. For PIT entropy, higher is better.



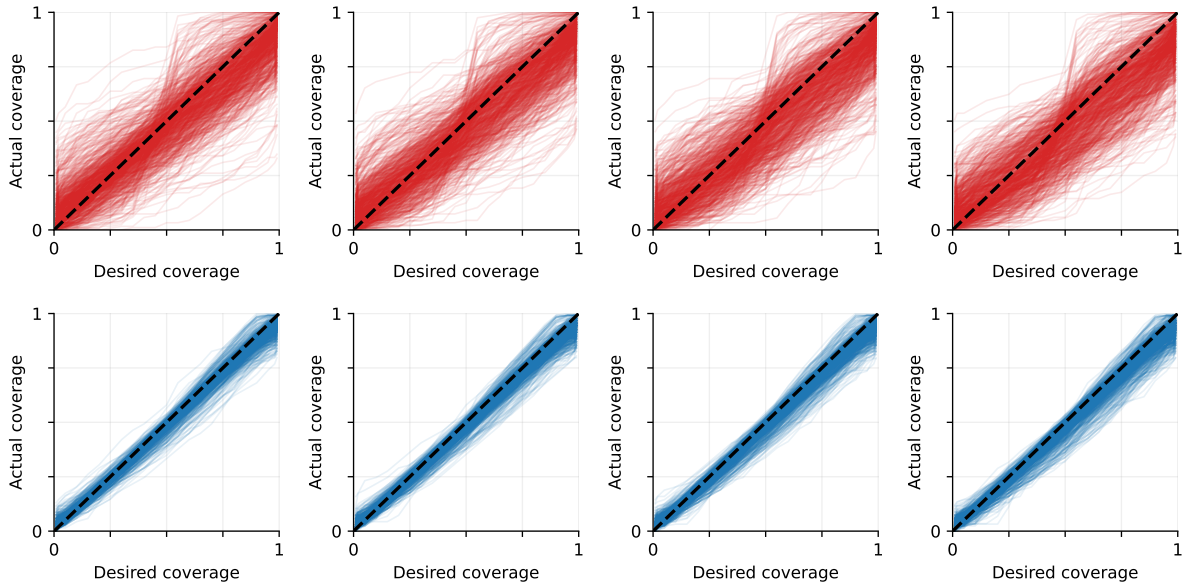
(a) Wind

(b) Solar

Figure 13: PIT entropy versus quantile loss on the energy dataset, analogous to Figure 7. For PIT entropy, higher is better.

Figures 15–18 display individual COVID-19 death forecasts before and after applying MultiQT, analogous to Figure 1 in the main text. To provide a sense of the effect of MultiQT on forecasts for states having large and small populations, the first pair of figures (Figures 15 and 16) show the effect of using MultiQT to correct one-week-ahead forecasts for California (the largest state by population) from each of the forecasting teams, and the second pair of figures (Figures 17 and 18) show the same for Vermont (one of the smallest states).

Recall that forecasts are made at levels 0.01, 0.025, 0.05, 0.1, 0.15, 0.2, 0.25, 0.3, 0.35, 0.4, 0.45, 0.5, 0.55, 0.6, 0.65, 0.7, 0.75, 0.8, 0.85, 0.9, 0.95, 0.975, and 0.99. To visualize these, we plot colored bands where the lightest opacity connects the 0.01 and 0.99 level forecasts, the next lightest connects the 0.025 and 0.975 level forecasts, and so on. We can use these plots to inspect calibration—if the 0.01 and 0.99 level quantile forecasts are calibrated, then we should see that the true value falls within the lightest opacity band 98% of the time. Zooming in on the raw forecasts, we can see that this is not the case for many of the forecasters initially, but after applying MultiQT the coverage of the extreme quantiles is much improved.



(a) $h = 1$ (b) $h = 2$ (c) $h = 3$ (d) $h = 4$

Figure 14: Actual versus desired coverage for COVID-19 death forecasting, analogous to Figure 4. Here the top row represents base forecasts, and the bottom row the forecasts after applying MultiQT.

E.3 Additional energy forecasting results

To complement Figure 6 from before, which shows calibration curves for daily energy forecasts at 10:00 a.m., we provide analogous plots for 2:00 a.m., 6:00 a.m., 2:00 p.m., 6:00 p.m., and 10:00 p.m., in Figure 19 (wind energy) and Figure 20 (solar energy). MultiQT produces near-perfect calibration at all hours. We note that in practice, it would be unnecessary to generate solar energy forecasts for 2:00 a.m. and 10:00 p.m. At these nighttime hours, the solar energy production is always zero, and the raw quantile forecasts are also zero for all levels. Figures 21 and 22 visualize the forecasts before and after applying MultiQT for the 10:00 a.m. time block at eight randomly sampled wind and solar farm sites.

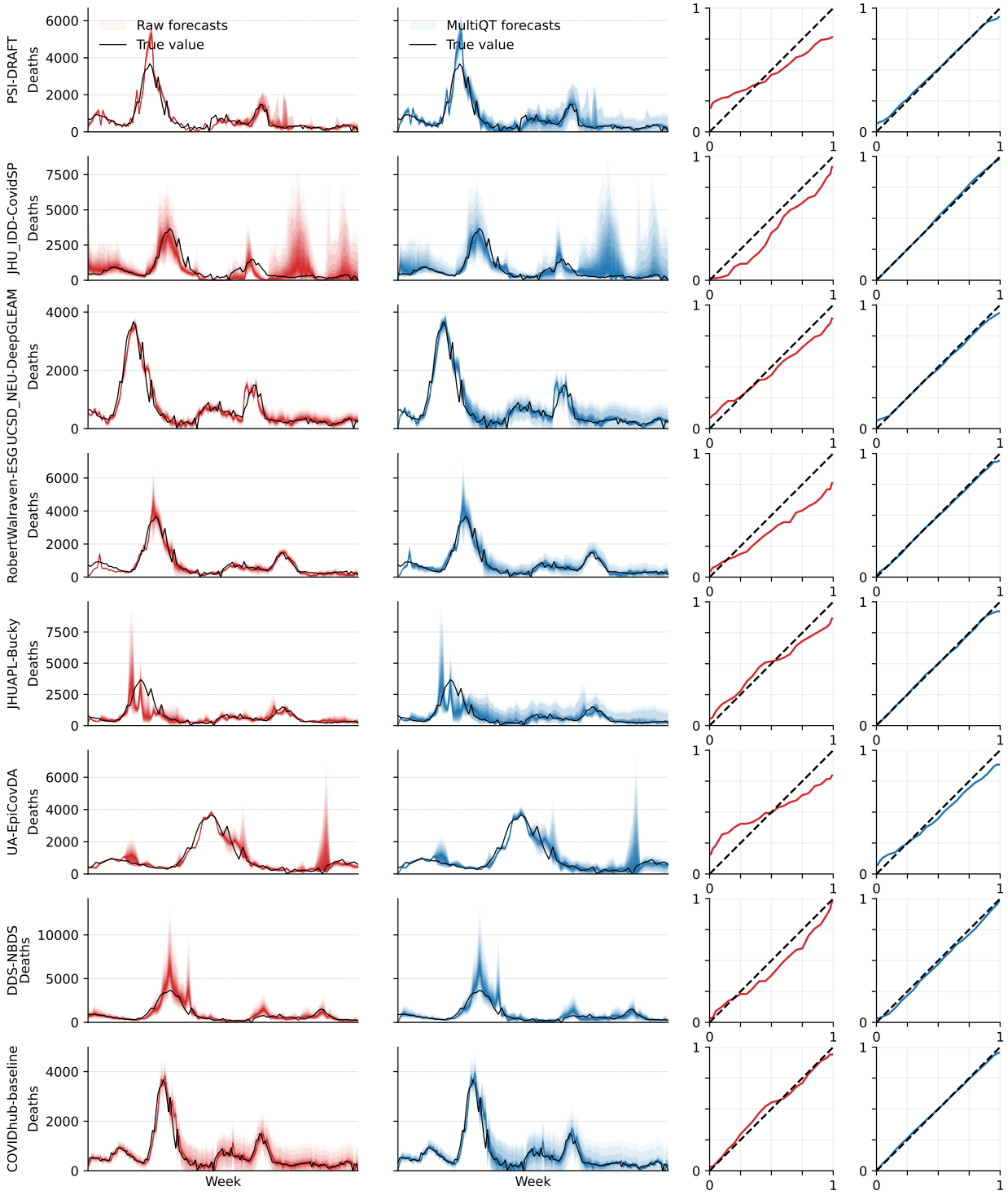


Figure 15: One-week-ahead forecasts of weekly COVID-19 deaths in California (part 1 of 2). Each row corresponds to one forecaster, each with their own forecast date range. The first column shows the raw forecasts, the second column shows the forecasts after using MultiQT, the third column shows actual versus desired coverage for the raw forecasts, and the fourth shows the same for the MultiQT forecasts.

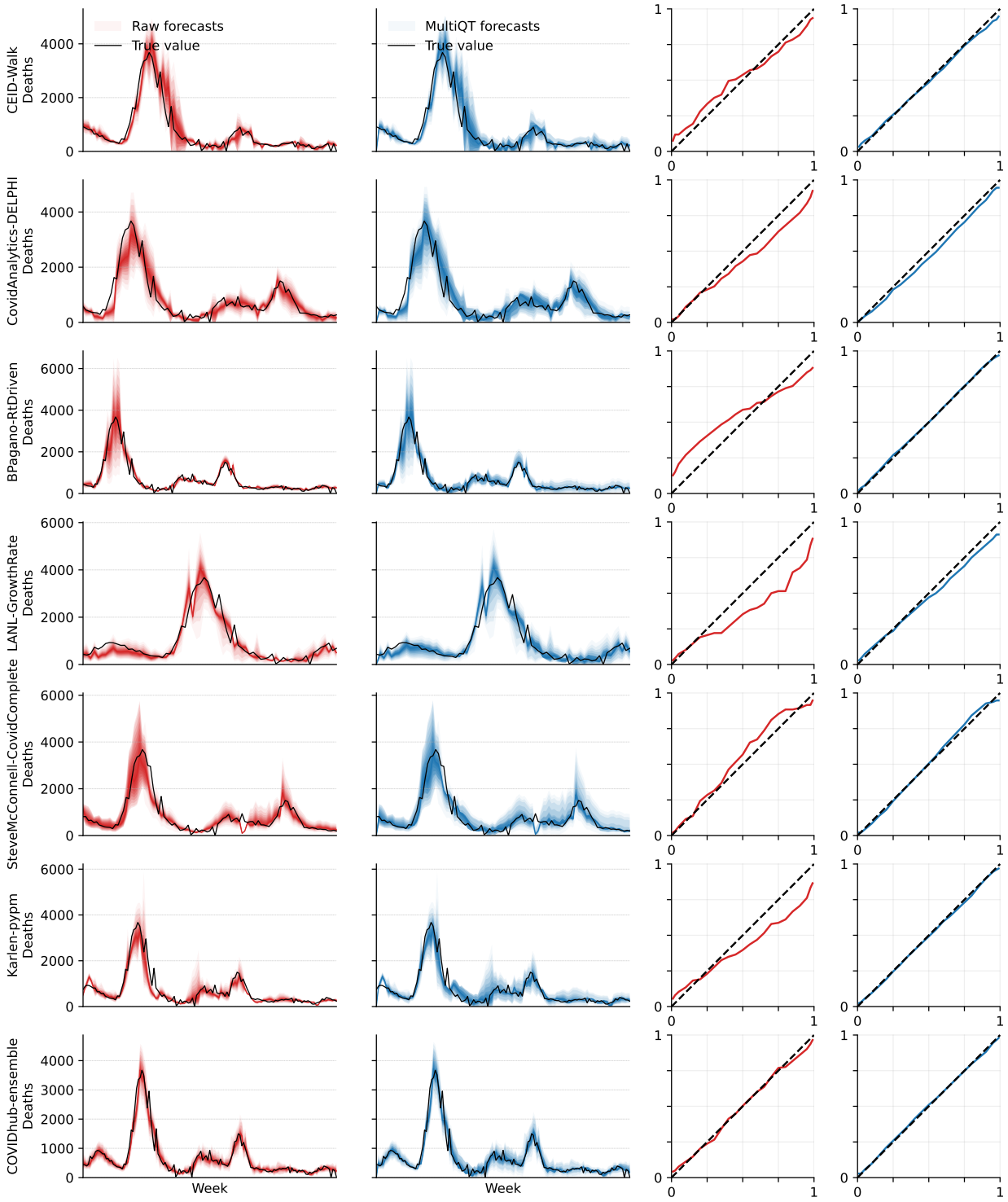


Figure 16: One-week-ahead forecasts of weekly COVID-19 deaths in California (part 2 of 2), as in Figure 15, for the remaining forecasters.

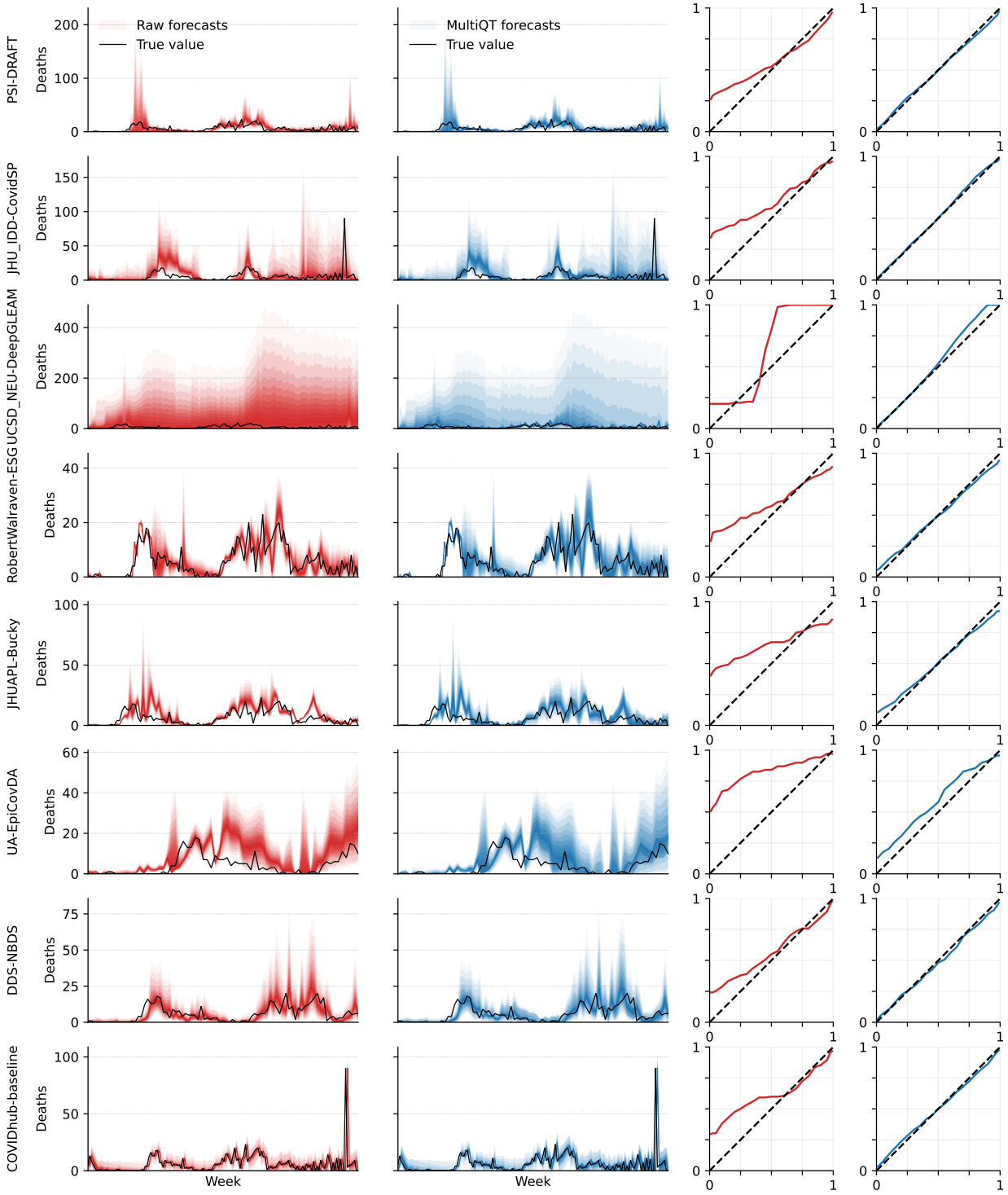


Figure 17: One-week-ahead forecasts of weekly COVID-19 deaths in Vermont (part 1 of 2). This is analogous to Figure 15, but for Vermont.

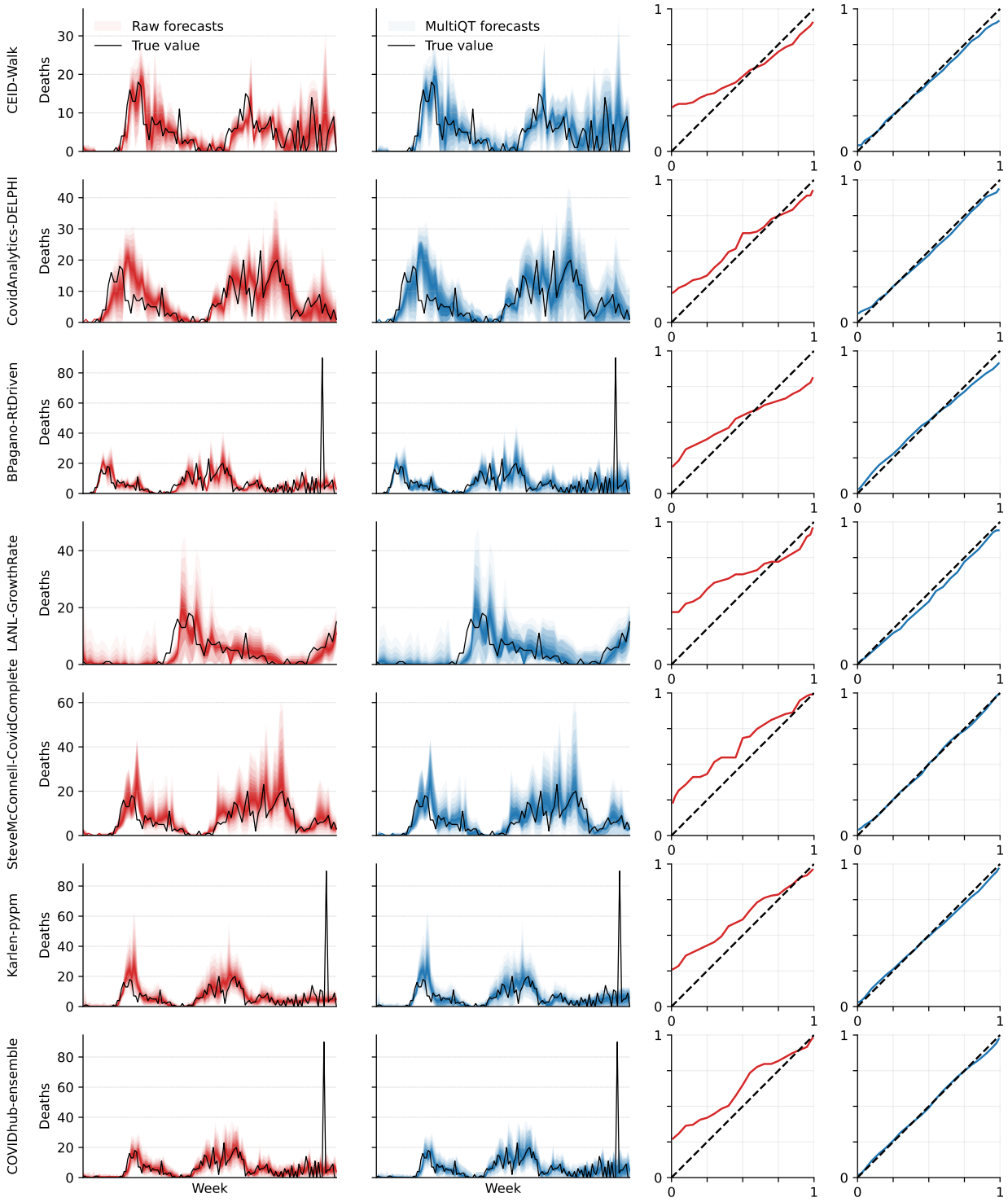
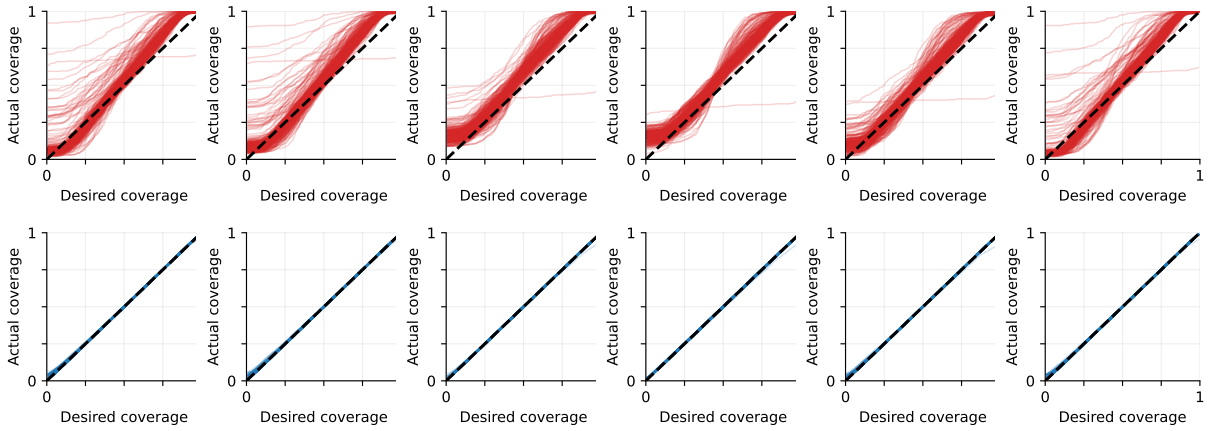
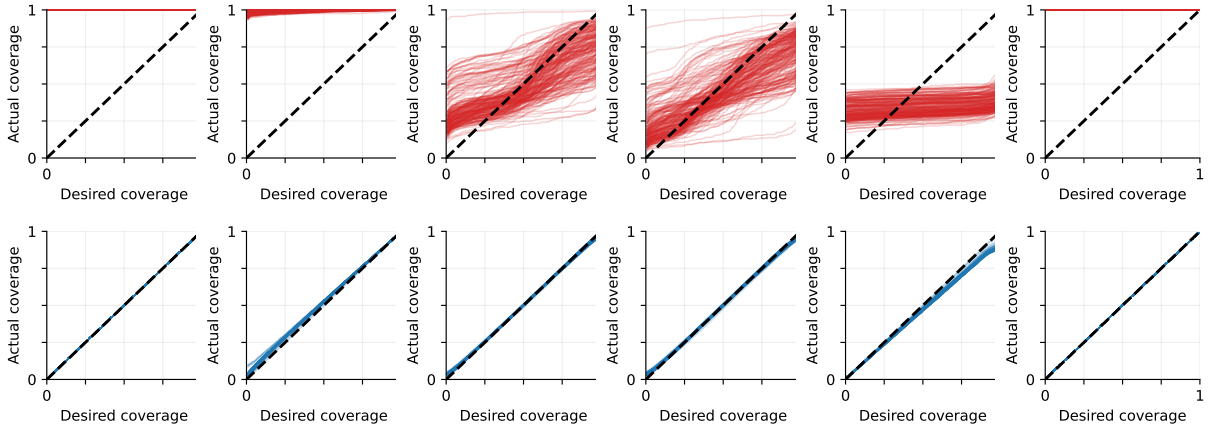


Figure 18: One-week-ahead forecasts of weekly COVID-19 deaths in Vermont (part 2 of 2), as in Figure 17, for the remaining forecasters.



(a) 2:00 a.m. (b) 6:00 a.m. (c) 10:00 a.m. (d) 2:00 p.m. (e) 6:00 p.m. (f) 10:00 p.m.

Figure 19: Actual versus desired coverage for wind energy forecasting, analogous to Figure 6. Here the top row corresponds to the raw forecasts, and the bottom row corresponds to the forecasts after applying MultiQT.



(a) 2:00 a.m. (b) 6:00 a.m. (c) 10:00 a.m. (d) 2:00 p.m. (e) 6:00 p.m. (f) 10:00 p.m.

Figure 20: Actual versus desired coverage for solar energy forecasting, analogous to Figure 6. Here the top row corresponds to the raw forecasts, and the bottom row corresponds to the forecasts after applying MultiQT.

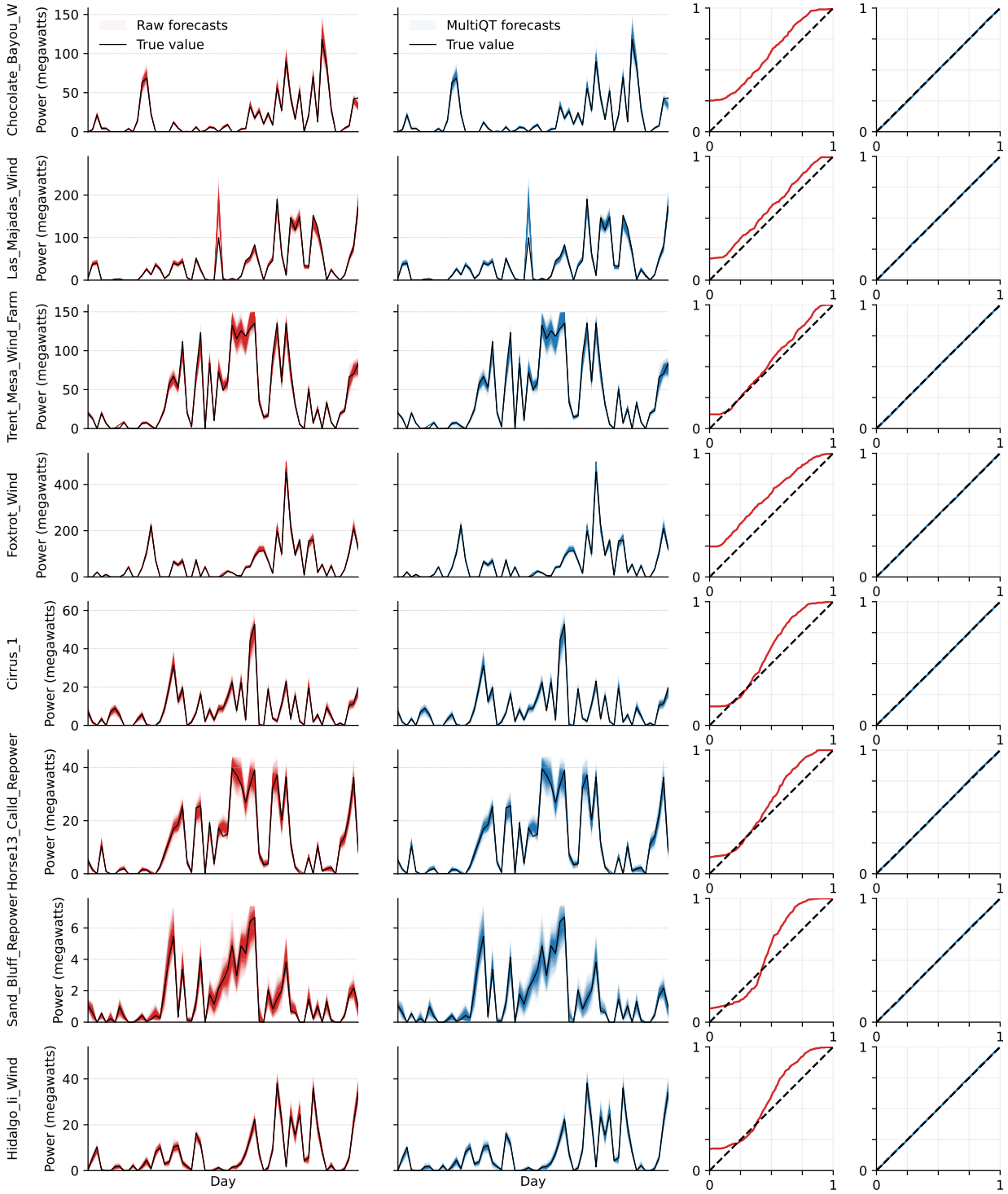


Figure 21: Day-ahead wind energy forecasts for the 10:00 a.m. time block at eight randomly sampled wind farm sites, where each row is a different site. For the sake of illustration, forecasts are plotted only for September 1, 2018 to October 31, 2018, but calibration is computed using forecasts for every day in 2018.

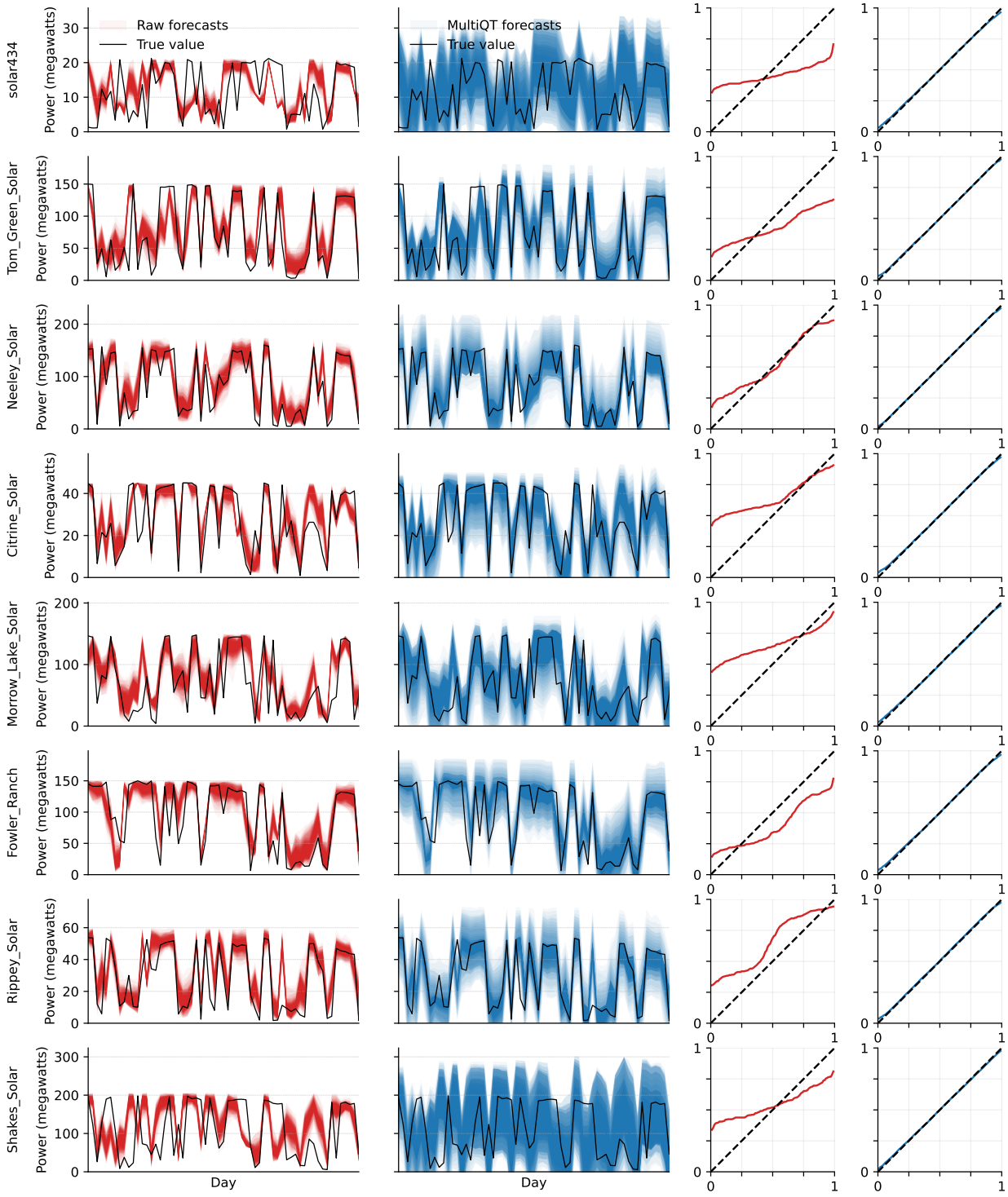


Figure 22: As in Figure 21, now for solar energy forecasting.