# Boosted Conformal Prediction Intervals

**Ran Xie**
Department of Statistics
Stanford University
ranxie@stanford.edu

**Rina Foygel Barber**
Department of Statistics
University of Chicago
rina@uchicago.edu

**Emmanuel J. Candès**
Department of Statistics
Department of Mathematics
Stanford University
candes@stanford.edu

## Abstract

This paper introduces a *boosted conformal procedure* designed to tailor conformalized prediction intervals toward specific desired properties, such as enhanced conditional coverage or reduced interval length. We employ machine learning techniques, notably gradient boosting, to systematically improve upon a predefined conformity score function. This process is guided by carefully constructed loss functions that measure the deviation of prediction intervals from the targeted properties. The procedure operates post-training, relying solely on model predictions and without modifying the trained model (e.g., the deep network). Systematic experiments demonstrate that starting from conventional conformal methods, our boosted procedure achieves substantial improvements in reducing interval length and decreasing deviation from target conditional coverage.

## 1 Introduction

Black-box machine learning algorithms have been increasingly employed to inform decision-making in sensitive applications. For instance, deep convolutional neural networks have been applied to diagnose skin cancer [14], and AlphaFold has been utilized in the development of malaria vaccines [24, 25]; here, scientists have employed AlphaFold to predict the structure of a key protein in the malaria parasite, facilitating the identification of potential binding sites for antibodies that could prevent the transmission of the parasite [25]. These instances highlight the critical need for understanding prediction accuracy, and one popular approach to quantify the uncertainty associated with general predictions relies on the construction of prediction sets guaranteed to contain the target label or response with high probability. Ideally, we would like the coverage to be valid conditional on the values taken by the features of the predictive model (e.g., patient demographics).

Conformal prediction [3] stands out as a flexible calibration procedure that provides a wrapper around any black-box prediction model to produce valid prediction intervals. Imagine we have a data set $\{(X_i, Y_i)\}_{i=1}^n$ and a test point $(X_{n+1}, Y_{n+1})$ drawn exchangeably from an unknown, arbitrary distribution $P$ (e.g. the pairs $(X_i, Y_i)$ may be i.i.d.). Taking the data set and the observed features $X_{n+1}$ as inputs, conformal prediction forms a prediction interval $C_n(X_{n+1})$ for $Y_{n+1}$ with valid marginal coverage, i.e. such that $\mathbb{P}(Y_{n+1} \in C_n(X_{n+1})) = 0.95$ or any nominal level specified by the user ahead of time. This is achieved by means of a conformity score $E(x, y; f)$, where $(x, y)$ represents a data point while $f$ represents any aspects of the distribution that we have estimated. For instance, the score may be given by the magnitude of the prediction error $|y - \hat{\mu}(x)|$, where $\hat{\mu}(x)$ represents the model prediction of the expected outcome, in which case $f$ is simply $\hat{\mu}$. Roughly, we

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

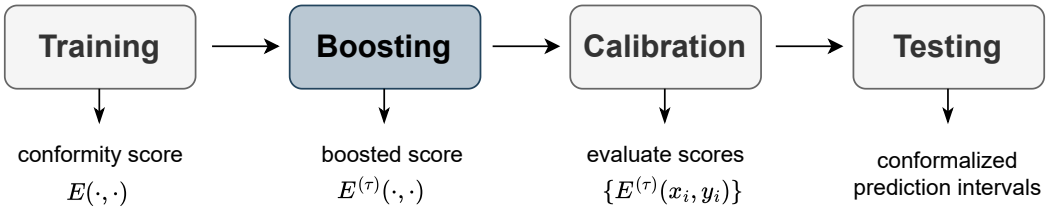



Figure 1: Illustration of the boosted conformal prediction procedure. We introduce a boosting stage between training and calibration, where we boost $\tau$ rounds on the conformity score function $E(\cdot,\cdot)$ and obtain the boosted score $E^{(\tau)}(\cdot,\cdot)$. The number of boosting rounds $\tau$ is selected via cross validation. A detailed description of the procedure is presented in Algorithm 1.

would include $y$ in the prediction interval if $E(X_{n+1}, y; f)$ does not take on an atypical value when compared with $\{E(X_i, Y_i; f)\}$, $i = 1, \ldots, n$. Selecting an appropriate conformity score is akin to choosing a test statistic in statistical testing, where two statistics may yield the same Type I error rate yet differ substantially in other aspects of performance.

One central issue is that while the conformal procedure guarantees marginal coverage, it does not extend similar guarantees to other desirable inferential properties without additional assumptions. In response, researchers have introduced a variety of conformity scores, including the locally adaptive (Local) conformity score [16], the conformalized quantile regression (CQR) conformity score [18], and its variants, CQR-m [22] and CQR-r [23]. Among these, CQR has often demonstrated superior empirical performance in terms of both interval length and conditional coverage [18].

This paper introduces a boosting procedure aimed at enhancing an arbitrary score function.[1] By employing machine learning techniques, namely, gradient boosting, our objective is to modify the Local or CQR score functions (or other baselines) to reduce the average length of prediction intervals or improve conditional coverage while maintaining marginal coverage. While this paper focuses primarily on length and conditional coverage, our methods can be tuned to optimize other criteria; we elaborate on this in Section 7.

Our boosted conformal procedure searches within a family of generalized scores for a score achieving a low value of a loss function adapted to the task at hand. Specifically, to evaluate the conditional coverage of prediction intervals, we build a loss function that maximizes deviation from the target coverage rate in the leaves of a shallow contrast tree [21]. Searching within a strategically designed family of score functions, rather than directly retraining or fine-tuning the fitted model under the task-specific loss function, yields greater flexibility and avoids the costs associated with retraining or fine-tuning. Further, this boosting process is executed post-model training, requiring only the model predictions and no direct access to the training algorithm.

Source code for implementing the boosted conformal procedure is available online at `https://github.com/ran-xie/boosted-conformal`. Details regarding the acquisition and preprocessing of the real datasets are also provided in the GitHub repository.

## 2 The split conformal procedure

We begin by outlining the key steps of the split conformal procedure applied to a family $\{(X_i, Y_i)\}_{i=1}^n$ of exchangeable samples (e.g., i.i.d.).

- *Training.* Randomly partition $[n]$ into a training set $I_1$ and a calibration set $I_2$. On the training set, train a model by means of an algorithm $\mathbb{A}$ to produce a conformity score function $E(\cdot, \cdot; f)$. The structure of this score function is predetermined, whereas the model $f$ is learned from $\mathbb{A}$. An example of a conformity score is $E(x, y; f) = |y - \hat{\mu}(x)|$, where $\hat{\mu}(x)$ is a learned regression function so that $f$ is here simply $\hat{\mu}$.
- *Calibration.* Evaluate the function $E(\cdot, \cdot; f)$ on each instance in the calibration set and obtain scores $\{E_i\}_{i \in I_2}$,[2] with each $E_i = E(X_i, Y_i; f)$. The $(1 - \alpha)$th empirical quantile

[1] An implementation of the boosted conformal procedure (BoostedCP) is available online at `https://github.com/ran-xie/boosted-conformal`.

[2] The term 'score' will henceforth refer to the conformity score unless stated otherwise.

of the score, $Q_{1-\alpha}(E, I_2)$, is calculated as

$$Q_{1-\alpha}(E, I_2) = \inf\{z : \mathbb{P}(Z \leq z) \geq 1 - \alpha\},$$

where $Z$ follows the distribution $\frac{1}{|I_2|+1}(\delta_\infty + \sum \delta_{E_i})$, and $\delta_a$ is a point mass at $a$.

- *Testing.* For a new observation $X_{n+1}$, output the conformalized prediction interval

$$C_n(X_{n+1}) = \{y \in \mathbb{R} : E(X_{n+1}, y; f) \leq Q_{1-\alpha}(E, I_2)\}. \tag{1}$$

If ties between $\{E_i\}_{i \in I_2}$ occur with probability zero, it holds that

$$1 - \alpha \leq \mathbb{P}(Y_{n+1} \in C_n(X_{n+1})) \leq 1 - \alpha + \frac{1}{|I_2| + 1}, \tag{2}$$

see [16]. By introducing additional randomization during the calibration step, the prediction interval can be tuned to obey $\mathbb{P}(Y_{n+1} \in C_n(X_{n+1})) = 1 - \alpha$, see [4]. This adjustment is not critical here and we omit the details.

**Locally adaptive conformal prediction** (Local for short) [16] introduces a score function that aims to make conformal prediction adapt to situations where the spread of the distribution of $Y$ varies significantly with the observed features $X$. On the training set, run an algorithm $\mathbb{A}$ to fit two functions $\mu_0(\cdot)$ and $\sigma_0(\cdot)$, where $\mu_0(X)$ estimates the conditional mean $\mathbb{E}[Y \mid X]$, and $\sigma_0(X)$ the dispersion around the conditional mean, frequently chosen as the conditional mean absolute deviation (MAD), $\mathbb{E}[|Y - \mu_0(X)| \mid X]$. With $f = (\mu_0, \sigma_0)$, the locally adaptive (Local) score function is:

$$E(x, y; f) = |y - \mu_0(x)|/\sigma_0(x). \tag{3}$$

For a new observation $X_{n+1}$, the conformalized prediction interval (1) takes on the simplified expression $[\mu_0(X_{n+1}) - Q_{1-\alpha}(E, I_2)\sigma_0(X_{n+1}), \mu_0(X_{n+1}) + Q_{1-\alpha}(E, I_2)\sigma_0(X_{n+1})]$.

**Conformalized quantile regression** (CQR) [17] also aims to adapt to heteroskedasticity by calibrating conditional quantiles, which often results in shorter prediction intervals. Apply quantile regression to produce a pair of estimated quantiles $(\hat{q}_{\alpha/2}(x), \hat{q}_{1-\alpha/2}(x))$, where $\hat{q}_\beta(X)$ is the estimated $\beta$th quantile of the conditional distribution of $Y$. The CQR score function is defined as

$$E(x, y; f) = \max\{\hat{q}_{\alpha/2}(x) - y, y - \hat{q}_{1-\alpha/2}(x)\}, \tag{4}$$

where $f = (\hat{q}_{\alpha/2}, \hat{q}_{1-\alpha/2})$. For a new observation $X_{n+1}$, following (1) yields the prediction interval

$$\left[\hat{q}_{\alpha/2}(X_{n+1}) - Q_{1-\alpha}(E, I_2), \hat{q}_{1-\alpha/2}(X_{n+1}) + Q_{1-\alpha}(E, I_2)\right]. \tag{5}$$

**Generalized conformity score families**. To construct a Local conformity score, we estimate two functions $\mu_0(\cdot)$ and $\sigma_0(\cdot)$ to plug into (3). Since these components are constructed without looking at performance downstream, it is reasonable to imagine that other choices may enjoy enhanced properties. How then should we systematically select $\mu(\cdot)$ and $\sigma(\cdot)$? To address this, we define a generalized Local score family $\mathcal{F}$ containing all potential score functions of the form

$$\mathcal{F} := \{E(\cdot, \cdot; f) : E(x, y; f) = |y - \mu(x)|/\sigma(x), \sigma(\cdot) > 0\}, \tag{6}$$

where $f = (\mu, \sigma)$. For each $E(\cdot, \cdot; f) \in \mathcal{F}$, the conformalized prediction interval is given by

$$\left[\mu(X) - Q_{1-\alpha}(E, I_2)\sigma(X), \mu(X) + Q_{1-\alpha}(E, I_2)\sigma(X)\right]. \tag{7}$$

Turning to CQR, one notable limitation is the uniform adjustment of prediction intervals by the constant factor $Q_{1-\alpha}(E, I_2)$, as shown in (5). This approach is suboptimal in the presence of heteroskedasticity, as it applies an identical correction to prediction intervals of varying widths for each $X = x$. Thus, simply updating the fitted quantiles $(\hat{q}_\alpha, \hat{q}_{1-\alpha/2})$ and plugging them into the original score function would be inadequate, as the structure of the original score imposes significant limitations on the effectiveness of conformalized prediction intervals. To address this, several variants including CQR-m [22] and CQR-r [23] have been proposed. Focusing on CQR-r, it employs a flexible score function, defined as $E(x, y; f) = \max\{\hat{q}_{\alpha/2}(x) - y, y - \hat{q}_{1-\alpha/2}(x)\}/(\hat{q}_{1-\alpha/2}(x) - \hat{q}_{\alpha/2}(x))$, with $f = (\hat{q}_{\alpha/2}, \hat{q}_{1-\alpha/2}, \hat{q}_{1-\alpha/2} - \hat{q}_{\alpha/2})$. Following (1), conformalized prediction intervals become

$$\left[\hat{q}_{\alpha/2}(X) - \hat{\sigma}(X)Q_{1-\alpha}(E, I_2), \hat{q}_{1-\alpha/2}(X) + \hat{\sigma}(X)Q_{1-\alpha}(E, I_2)\right], \tag{8}$$

where $\hat{\sigma} = \hat{q}_{1-\alpha/2} - \hat{q}_{\alpha/2}$. Intuitively, the adjusted score function allows prediction bands to adjust in proportion to their width, instead of adding a constant shift as in CQR. However, despite the intuitive appeal of adjusted scores as a seemingly more reasonable "allocation" of the conformal correction, empirical studies reveal that they do not result in narrower prediction intervals when compared to CQR [23]. This phenomenon is largely due to the uniform direction of the conformal adjustment, represented by $Q_{1-\alpha}(E, I_2)$, across all observations. In particular, if $Q_{1-\alpha}(E, I_2) < 0$, indicating that the true target $y$ predominantly lies within the estimated quantile range $[\hat{q}_{\alpha/2}, \hat{q}_{1-\alpha/2}]$, there is a uniform narrowing of the predicted interval across all samples.

In light of these insights, we propose a novel score family, $\mathcal{H}$, designed to augment the flexibility of the conformity score functions:

$$\mathcal{H} := \left\{ E(\cdot, \cdot; f) : E(x, y; f) = \max\left\{\mu_1(x) - y, y - \mu_2(x)\right\} / \sigma(x), \mu_1(\cdot) \leq \mu_2(\cdot), \sigma(\cdot) > 0 \right\}, \quad (9)$$

where $f = (\mu_1, \mu_2, \sigma)$, which leads to conformalized prediction intervals of the form

$$[\mu_1(X) - \sigma(X) Q_{1-\alpha}(E, I_2), \mu_2(X) + \sigma(X) Q_{1-\alpha}(E, I_2)]. \quad (10)$$

Notably, $\mathcal{H}$ includes the Local, CQR, and CQR-r scores as special cases.

## 3 Boosted conformal procedure

It is clear from above that a model is trained to produce a conformity score $E(\cdot, \cdot; f)$; e.g., we may learn a regression function $\hat{\mu}(\cdot)$ to plug it into a score function $|y - \hat{\mu}(x)|$. To overcome the limitation of working with an arbitrarily selected score function, we introduce a boosting step before calibration, see Figure 1. In a nutshell, we use gradient boosting to iteratively improve upon a predefined score $E(\cdot, \cdot; f)$ now denoted as $E^{(0)}(\cdot, \cdot)$, where the superscript indicates the 0th iteration.

To achieve this, we construct a task-specific loss function $\ell$, which takes a dataset $\mathcal{D}$ and a score function $E(\cdot, \cdot; f)$ as inputs, and outputs $\ell(E(\cdot, \cdot; f); \mathcal{D})$ measuring how closely the conformalized prediction interval aligns with the analyst's objective. This loss function $\ell$ is designed to be differentiable with respect to each of the model components produced by the training algorithm. Importantly, it does not require knowledge of the gradient of $f(x)$ with respect to $x$. In the example above, taking the labels as fixed, this means that for each feature $x_i \in \mathcal{D}$, $i = 1, \ldots, n$, if we set $\hat{y}_i = \hat{\mu}(x_i)$, then the loss $\ell(E(\cdot, \cdot); \mathcal{D})$ is a function of $\{\hat{y}_i\}_{i=1}^n$, and the derivative $\partial \ell(E(\cdot, \cdot); \mathcal{D}) / \partial \hat{y}_i$ is well defined. In Sections 5.1 and 6.1, we present examples of such derivatives.

Each boosting iteration updates the score function sequentially, employing a gradient boosting algorithm such as XGBoost [12] or LightGBM [15]. These algorithms accept as input a dataset $\mathcal{D}$, a base score function $E(\cdot, \cdot; f)$, a custom loss function $\ell$, gradients of $\ell$ with respect to $f$ (denoted $\nabla_f \ell$), and a number of boosting rounds $\tau$. We may write the boosting procedure as

$$(E^{(0)}(\cdot, \cdot), \ldots, E^{(\tau)}(\cdot, \cdot)) = \text{GradientBoosting}(\mathcal{D}, E(\cdot, \cdot; f), \ell, \nabla_f \ell, \tau). \quad (11)$$

This yields a boosted score function $E^{(\tau)}(\cdot, \cdot)$, which is then used for calibration and for constructing prediction intervals. The number $\tau$ is calculated using $k$-fold cross-validation on the training dataset, selecting $\tau$ from potential values up to a predefined maximum $T$ (e.g., 500). We partition the dataset into $k$ folds and for each $j = 1, \ldots, k$, we hold out fold $j$ for sub-calibration and the remaining $k - 1$ folds for sub-training. We apply $T$ rounds of gradient boosting (11) on the sub-training data, generating $T + 1$ candidate score functions $E_j^{(0)}(\cdot, \cdot), \ldots, E_j^{(T)}(\cdot, \cdot)$. Each score function is then evaluated on sub-calibration data, using the loss function $\ell$ to compute losses at all epochs, i.e., for each fold $j = 1, \ldots k$,

$$\{L_j^{(t)}\}_{t=0}^T = \{\ell(E_j^{(t)}; \text{fold}_j)\}_{t=0}^T.$$

Last, $\tau$ is selected as the round that minimizes the average loss across all $k$ folds:

$$\tau = \arg\min_{0 \leq t \leq T} \textstyle\sum_{j=1}^k L_j^{(t)}, \quad (12)$$

see Figure 2. This cross-validation step simulates the calibration step in conformal prediction and effectively prevents the overfitting of the score function.

Since boosting is conducted on the training data, the boosted procedure satisfies the same marginal coverage guarantee as the split conformal procedure, as formalized below.

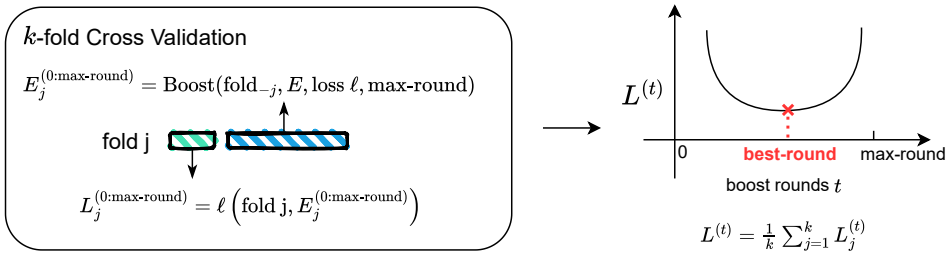



Figure 2: Schematic drawing showing the selection of the number of boosting rounds via cross-validation. Left: we hold out fold $j$, and use the remaining $k-1$ folds to generate candidate scores $E_j^{(t)}$, $t = 0, \ldots,$ max-round. The performance of each score is evaluated on fold $j$ using the loss function $\ell$. Right: best-round minimizes the average loss across all $k$ folds. A detailed description of the procedure is presented in Algorithm 1.

**Proposition 3.1.** *Let $\{(X_i, Y_i)\}_{i=1}^m$ be the held out calibration set, and $(X_{m+1}, Y_{m+1})$ be a pair of new observation. If the $m+1$ samples are exchangeable, and ties between $\{E^{(\tau)}(X_i, Y_i)\}_{i=1}^m$ occur with probability zero, the confromalized prediction interval (1) computed from score function $E^{(\tau)}(\cdot,\cdot)$ satisfies the coverage guarantee (2).*

**Searching within generalized conformity score families**. To update the Local score function (3), we search within the generalized score family $\mathcal{F}$ (6). First, we initialize $\mu^{(0)} = \mu_0$ and $\sigma^{(0)} = \sigma_0$. After completing $\tau$ iterations of boosting on the training set, we obtain the boosted score function $E^{(\tau)}(x, y) = |y - \mu^{(\tau)}(x)|/\sigma^{(\tau)}(x)$. Notably, we can update any score function within $\mathcal{F}$. For instance, to update $E(x, y; f) = |y - \hat{\mu}(x)|$, we simply initialize $\mu^{(0)} = \hat{\mu}$, and take $\sigma^{(0)}$ to be the constant function equal to one. Similarly, to update the CQR score function (4), we search within the score family $\mathcal{H}$ (9). First, we initialize a triple $\mu_1^{(0)} = \hat{q}_{\alpha/2}$, $\mu_2^{(0)} = \hat{q}_{1-\alpha/2}$, $\sigma^{(0)} = \hat{q}_{1-\alpha/2} - \hat{q}_{\alpha/2}$. After $\tau$ boosting rounds, we obtain the boosted score function $E^{(\tau)}(x, y) = \max\{\mu_1^{(\tau)}(x) - y, y - \mu_2{}^{(\tau)}(x)\}/\sigma^{(\tau)}(x)$.

**Algorithm 1** Boosting stage

**Input:**

Training data $(X_i, Y_i) \in \mathbb{R}^p \times \mathbb{R}$, $i = 1, ..., n$; base conformity score function $E^{(0)}(\cdot,\cdot)$
Loss function $\ell$; target mis-coverage level $\alpha \in (0, 1)$
Number $k$ of cross-validation folds; maximum boosting rounds $T$

**Procedure:**

Randomly divide $\{1, ..., n\}$ into $k$ folds
**for** $j \leftarrow 1$ `to` $k$ **do**
  Set fold $j$ as sub-calibration set, and the remaining $k-1$ folds as sub-training set
  On the sub-training set, call GradientBoosting (11) to obtain candidate scores $\{E_j^{(t)}\}_{t=0}^T$
  On the sub-calibration set, evaluate $L_j^{(t)} = \ell(E_j^{(t)})$, $t = 0, \ldots, T$
**end for**
Set boosting rounds $\tau \leftarrow \arg\min_t \frac{1}{k}\sum_{j=1}^k L_j^{(t)}$ as in (12)
On the training set, call GradientBoosting (11) to obtain boosted functions $\{E^{(t)}\}_{t=0}^\tau$

**Output:**

Boosted conformity score function $E^{(\tau)}(\cdot,\cdot)$

## 4 Related Works

**Adapting the classical conformal procedure** to improve properties of the conformalized intervals has been one of the primary focuses of recent literature. Noteworthy contributions—including CF-GNN [28] and ConTr [26]—approach this problem by introducing modifications to the training stage

of the procedure. As outlined in Section 2, a model is trained to produce a score function $E(\cdot, \cdot; f)$. The model $f$ usually depends on a set of model parameters, e.g., neural network parameters $\theta$. Denote the trained model $f$ by $f_\theta$. CF-GNN and ConTr retrain or fine-tune the model by using a carefully constructed loss function, which may aim to produce narrower prediction intervals or prediction sets of reduced cardinality in classification problems. This process generates a new set of model parameters $\theta'$. The new model $f_{\theta'}$ is then plugged into the *same* predefined conformity score function—namely CQR [28] or the adaptive prediction set score (APS) [26]—to produce $E(\cdot, \cdot, f_{\theta'})$.

There are two primary limitations. First, the score function imposes constraints on the properties of conformalized intervals as explained in Section 2. Our approach introduces more flexibility by constructing a family of generalized score functions that is a superset of $\{E(\cdot, \cdot; f_\theta) : \theta \in \Theta\}$, where $\Theta$ is the parameter space of the training model. This family is strategically designed to contain an oracle conformity score ideally suited to the task at hand, e.g., achieving exact conditional coverage. Second, current methodologies necessitate fine-tuning or retraining models from scratch, requiring both access to the training model and significant computational resources. In contrast, our boosted conformal method operates directly on model predictions and circumvents these issues.

**Conditional coverage** of conformalized prediction intervals has also attracted significant interest, characterized by efforts to establish theoretical guarantees and achieve numerical improvements. Prior work established an impossibility result [8, 20], which states that exact conditional coverage in finite samples cannot be guaranteed without making assumptions about the data distribution. Subsequently, Gibbs et al. [27] developed a modified conformal procedure that guarantees conditional coverage for predefined protected sub-groups, i.e. subsets of the feature space. Our approach differs from the previous works by introducing a numerical method directly aimed at improving the conditional coverage, $\mathbb{P}(Y \in C_n(X)|X = x)$, across all potential values of $x$.

## 5 Boosting for conditional coverage

Maintaining valid marginal coverage, our goal is to produce a prediction interval $C_n$ obeying

$$\mathbb{P}(Y \in C_n(X_{n+1})|X_{n+1} = x) \approx 1 - \alpha \tag{13}$$

for all possible values of $x$. To this end, we present a loss function that quantifies the conditional coverage rate of any prediction interval. Requiring merely a dataset $\mathcal{D}$ and a prediction interval $C_n(\cdot)$ as inputs, it also serves as an effective evaluation metric, which may be of independent interest.

### 5.1 A measure for deviation from target conditional coverage

From now on, we let $E$ be the score function $E(\cdot, \cdot; f)$. Set $\mathcal{D} = \{(X_i, Y_i)\}_{i=1}^n$ and denote by $C_n(\cdot)$ the conformalized prediction interval constructed from $E$. We shall assess the deviation of $C_n(\cdot)$ from the target conditional coverage by means of Contrast Trees [21]. As background, a contrast tree iteratively identifies splits within the feature space $\mathcal{X}$ in a greedy fashion, aiming to maximize absolute within-group deviations from the target conditional coverage rate $(1-\alpha)$. For a subset $R$ of the data point indices $[n]$, let $\mathcal{D}_R = \{X_j, Y_j\}_{j \in R}$. The absolute within-group deviation is computed as

$$d\left(C_n(\cdot); \mathcal{D}_R\right) = \left| |R|^{-1} \textstyle\sum_{j \in R} \mathbb{1}(Y_j \in C_n(X_j)) - (1 - \alpha) \right|. \tag{14}$$

The overall empirical maximum deviation is then defined as

$$\ell_M\left(E; \mathcal{D}\right) = \max\nolimits_{1 \le m \le M} d\left(C_n(\cdot); \mathcal{D}_{\hat{R}_m}\right), \tag{15}$$

where $\hat{R}_1 \cup \cdots \cup \hat{R}_M$ is a partition of $[n]$, which itself depends on $E$ and $\mathcal{D}$. Specifically, it is computed by running a contrast tree for $M$ iterations. At each iteration, the algorithm not only seeks to isolate regions with large deviations but also discourages splits where any subset $\hat{R}_m$ is too small.

To update score functions via gradient boosting as described in (11), we would need a differentiable approximation of the maximum deviation. To this end, we construct approximations for the following three components of the loss function. With an abuse of notation, in subsequent discussions, we shall employ the same notations to denote these differentiable approximations.

1. Approximation for the prediction interval $C_n(\cdot)$ in (14): the prediction interval is formulated as (7) for the generalized Local score, and as (10) for the generalized CQR score.

Denote the upper and lower limits of $C_n(\cdot)$ by $u(\cdot)$ and $l(\cdot)$. We approximate the empirical quantile $Q_{1-\alpha}(E, I_2)$ in $u(\cdot)$ and $l(\cdot)$ with a smooth quantile estimator $Q^s_{1-\alpha}$. Given $r$ scalars $\{z_i\}_{i=1}^r$, $Q^s_{1-\alpha}$ is constructed as:

$$Q^s_{1-\alpha}(\{z_i\}_{i=1}^r) := \langle \text{HD}(r), s(\mathbf{z}) \rangle, \tag{16}$$

where $\langle \cdot, \cdot \rangle$ represents the dot product. Here, $\text{HD}(r) = [W_{r,1}, ..., W_{r,r}]$ is the weight vector corresponding to the Harrel-Davis distribution-free empirical quantile estimator [1], and $s(\mathbf{z})$ is a differentiable ordering $\{\tilde{z}_{(i)}\}_{i=1}^r$, arranged in the ascending order. In practice, the derivative of $s(\mathbf{z})$ with respect to each $z_i$ is given by the package developed in [19]. This approach is a smooth approximation of the Harrel-Davis quantile estimator $Q^{\text{HD}}_{1-\alpha}$, constructed as a linear combination of the order statistics, $Q^{\text{HD}}_{1-\alpha} = \langle \text{HD}(r), \{z_{(i)}\}) \rangle = \sum_{i=1}^r W_{r,i} z_{(i)}$, where $W_{r,i}$ takes the value $I_{(1-\alpha)(r+1),\alpha(r+1)}(i/r) - I_{(1-\alpha)(r+1),\alpha(r+1)}((i-1)/r)$ and $I_{a,b}(x)$ represents the incomplete beta function.

2. Approximation for absolute deviation $d_i$ (14): the indicator function in (14) can be approximated by the product of two sigmoid functions,

$$\begin{aligned} \mathbb{1}(Y_j \in C_n(X_j)) &= \mathbb{1}(u(X_j) - Y_j \geq 0)\mathbb{1}(Y_j - l(X_j) \geq 0) \\ &\approx S_{\tau_1}(u(X_j) - Y_j) S_{\tau_1}(Y_j - l(X_j)), \end{aligned}$$

where $\tau_1$ is a parameter, trading off smoothness and quality of the approximation. The sigmoid function $S_{\tau_1}(x)$ is defined as $S_{\tau_1}(x) = (1 + e^{-\tau_1 x})^{-1}$.

3. Approximation for maximum deviation: we employ a log-sum-exp function [2] to derive the differentiable approximation of $\ell_M$ as

$$\ell_M(E; \mathcal{D}) := {\tau_2}^{-1} \log \textstyle\sum_{m=1}^M \exp\left(\tau_2 d_m(C_n(\cdot); \mathcal{D}_m)\right), \tag{17}$$

where $\tau_2$ is a parameter, serving the same purpose as $\tau_1$.

Here, we demonstrate calculating the derivative of the smooth approximation (17) with respect to each component of the generalized Local score, expanding it as follows:

$$\ell_M(E; \mathcal{D}) = {\tau_2}^{-1} \log \sum_{m=1}^M \exp\left(\tau_2 \left| |R_m|^{-1} \textstyle\sum_{j \in R_m} S_{\tau_1}(u(X_j) - Y_j) S_{\tau_1}(Y_j - l(X_j)) - (1-\alpha) \right| \right),$$

where

$$\begin{aligned} S_{\tau_1}(u(X_j) - Y_j) &= \left(1 + \exp\left[-\tau_1(\mu_j + Q^s_{1-\alpha}(\{E_i\}_{i=1}^n)\sigma_j - Y_j)\right]\right)^{-1}, \\ S_{\tau_1}(Y_j - l(X_j)) &= \left(1 + \exp\left[-\tau_1(Y_j - \mu_j + Q^s_{1-\alpha}(\{E_i\}_{i=1}^n)\sigma_j)\right]\right)^{-1}, \end{aligned}$$

with $\mu_i = \mu(X_i)$, $\sigma_i = \sigma(X_i)$, $E_i = |Y_i - \mu_i|/\sigma_i$. As a result, for each feature $X_i$ within $\mathcal{D}$, we can evaluate $\partial \ell_M(E; \mathcal{D}) / \partial \mu_i$ and $\partial \ell_M(E; \mathcal{D}) / \partial \sigma_i$ via the chain rule.

### 5.2 Boosting score functions for conditional coverage

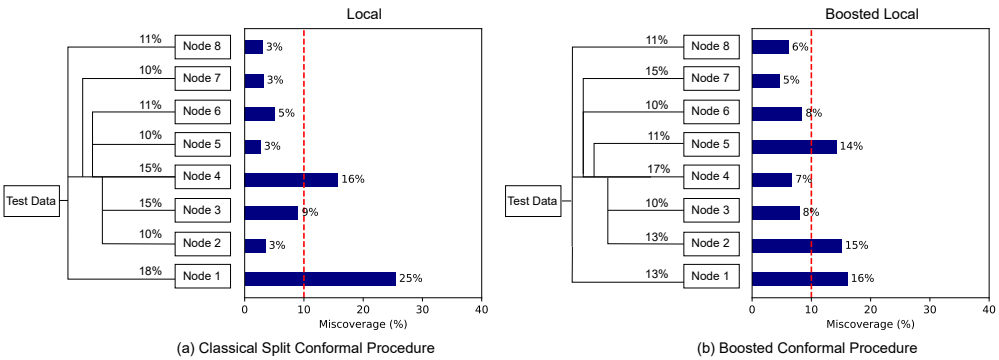
Since the empirical maximum deviation $\ell_M$ (15) is non-differentiable, we opt for the differentiable approximation during the gradient boosting step (11). Nonetheless, we utilize the original $\ell_M$ to select the number of boosting rounds as in step (12) and to evaluate the conditional coverage of the conformalized prediction interval on the test set.

#### 5.2.1 Theoretical guarantees

The oracle score function achieving conditional coverage as defined in (13) belongs to both proposed generalized score families.

**Proposition 5.1** (Asymptotic expressiveness)**.** *Let $\{X_i, Y_i\}_{i=1}^n$ be i.i.d. with continuous joint probability density distribution. Under the split conformal procedure, for any target coverage rate $1-\alpha$, as $n \to \infty$, there exists $(\mu^*, \sigma^*)$ and $(\mu_1^*, \mu_2^*, \sigma^*)$ such that the corresponding generalized Local (6) and CQR (9) score functions recover conditional coverage at rate $1-\alpha$, as defined in (13).*

It goes without saying that there is no reason to assume that the optimal $\mu^*$ corresponds to the conditional mean, median or any quantile of $Y$ given $X$, or that the optimal $\sigma^*$ corresponds to the standard deviation or the mean absolute deviation of $Y$ given $X$, as in the original Local score (3). That said, our greedy strategy has no guarantee on global optimality and this is why the choice of the starting point—whether it is the Local or CQR score function—plays a role in the performance.




(a) Classical Split Conformal Procedure (b) Boosted Conformal Procedure

Figure 3: Comparison of test set conditional coverage evaluated on the dataset meps-19: (a) shows the classical Local-type conformal procedure and (b) our boosted Local-type conformal procedure. The target miscoverage rate is set to $\alpha = 10\%$ (red). Miscoverage rate is computed at each leaf of the contrast tree, constructed to detect deviation from the target rate. Each leaf node is labeled with its size, namely, the fraction of the test set it represents.

Table 1: Test set maximum deviation loss $\ell_M$ evaluated on various conformalized intervals. The best result achieved for each dataset is highlighted in bold.

| Max. Conditional Coverage Deviation (%), target miscoverage $\alpha = 10\%$ | | | | | | |
|---|---|---|---|---|---|---|
| Dataset | Method | | Improvement | Method | | Improvement |
| | Local | Boosted | | CQR | Boosted | |
| bike | 10.979 | 5.638 | -48.65% | 4.934 | **4.925** | -0.17% |
| bio | 5.303 | 4.862 | -8.31% | 5.069 | **4.700** | -7.29% |
| community | 25.755 | 13.466 | -47.71% | 12.688 | **12.105** | -4.59% |
| concrete | 10.740 | 8.763 | -18.40% | 9.039 | **8.265** | -8.56% |
| meps-19 | 15.357 | 5.656 | -63.17% | 5.507 | **5.507** | -0.00% |
| meps-20 | 16.939 | **6.998** | -58.69% | 7.614 | 7.184 | -5.65% |
| meps-21 | 17.627 | **7.832** | -55.57% | 8.165 | 8.067 | -1.20% |

### 5.2.2 Empirical results on real data

We apply our boosted conformal procedure to the 11 datasets previously analyzed in [23, 18, 22]. Details on the datasets are provided in Section A.6 in the Appendix. In each dataset, we randomly hold out 20% as test data. All experiments are repeated 10 times, starting from the data splitting. We refer to Section A.7 for details on the models and hyper-parameters we employ for the training and boosting stages.

We evaluate the conditional coverage of the prediction intervals as the maximum within-group deviations across a partitioned test set (15). This partition is obtained through a contrast tree algorithm described in Section 5.1. Figure 3 illustrates the comparison between miscoverage rates of prediction intervals at each leaf of the contrast tree. These intervals are derived under the classical Local conformal procedure and our boosted conformal procedure. Notably, the conditional coverage of the boosted prediction interval more closely aligns with the target rate $1 - \alpha$.

The experiment results summarized in Table 1 indicate that applying boosting significantly enhances the performance of the baseline Local procedure. In contrast, boosting on CQR does not yield significant improvements—a sign that CQR already targets conditional coverage. (Before boosting, the prediction intervals generated by the baseline Local procedure exhibit conditional coverage deviations up to three times greater than those of the baseline CQR procedure.) It is noteworthy, however, that after boosting, the conditional coverage of the Local procedure improves to a level comparable to that of the boosted CQR procedure. While generally slightly less effective, nevertheless surpasses the performance of the boosted CQR procedure in two cases. Results on the remaining datasets are deferred to Tables A2 and A3.

# 6 Boosting for length

We begin by specifying the oracle prediction interval with minimum length. For a random variable $Z$, the High Density Region (HDR) at a specified significance level $\alpha$, denoted as $\text{HDR}_\alpha(Z)$, is defined as the shortest deterministic interval that covers $Z$ with probability at least $1-\alpha$. The boundaries of $\text{HDR}_\alpha(Z)$, the lower limit $Q_{l(\alpha)}$ and the upper limit $Q_{u(\alpha)}$, obey the condition $\mathbb{P}(Z \in [Q_{l(\alpha)}, Q_{u(\alpha)}]) \geq 1-\alpha$. For a pair of $(X, Y)$ drawn from $P$, for every value of $x \in \mathbb{R}^p$, the oracle prediction interval at that point is expressed as

$$\text{HDR}_\alpha(Y|X=x) = \left[Q_{l(\alpha)}(Y|X=x), Q_{u(\alpha)}(Y|X=x)\right]. \tag{18}$$

Before introducing the boosting strategy, we present a word of caution against optimizing exclusively for this objective. Importantly, to maintain valid marginal coverage, the shortest prediction interval is prone to overcover when the spread of $Y|X$ (the conditional distribution of $Y$ given $X$) is small, and undercover when the spread of $Y|X$ is large. This may be undesirable.

Similar to Proposition 5.1, we can show that the generalized score families exhibit the necessary expressiveness to contain the oracle conformity score, achieving optimal length while ensuring valid marginal coverage. The formal proof is deferred to Section A.3.

## 6.1 A measure for length

Consider a dataset $\mathcal{D} = \{(X_i, Y_i)\}_{i=1}^n$ and a score function $E$. Denote the corresponding conformalized prediction interval by $C_n(\cdot)$, with its quality measured by the average length:

$$\ell_L(E; \mathcal{D}) = n^{-1} \textstyle\sum_{i=1}^n |C_n(X_i)|. \tag{19}$$

To derive a differentiable approximation of $\ell_L$, we approximate the empirical quantile $Q_{1-\alpha}$ in the conformalized intervals (7) and (10) with the smooth quantile estimator $Q^s_{1-\alpha}$ constructed in (16). Here, we demonstrate calculating the derivative of the smooth approximation of $\ell_L$ with respect to each component of the generalized Local score, expanding it as follows based on the previously outlined approximation steps:

$$\ell_L\left(E; \mathcal{D}\right) = 2\bar{\sigma} Q^s_{1-\alpha}(\{E_i\}_{i=1}^n), \qquad E_i = |Y_i - \mu_i|/\sigma_i,$$

with $\mu_i = \mu(X_i)$, $\sigma_i = \sigma(X_i)$, $\bar{\sigma} = n^{-1}\sum_{i=1}^n \sigma_i$. As a result, for each feature $X_i$ within $\mathcal{D}$, we can evaluate $\partial \ell_L\left(E; \mathcal{D}\right)/\partial \mu_i$ and $\partial \ell_L\left(E; \mathcal{D}\right)/\partial \sigma_i$ via the chain rule. For instance,

$$\frac{\partial \ell_L\left(E; \mathcal{D}\right)}{\partial \mu_i} = -2\bar{\sigma} \frac{\partial Q^s_{1-\alpha}(\{E_j\}_{j=1}^n)}{\partial E_i} \frac{\text{sign}(Y_i - \mu_i)}{\sigma_i}.$$

## 6.2 Empirical results on real data

We apply our boosted conformal procedure to the same datasets described in Section 5.2.2. Detailed information on the models and hyperparameters used during the training and boosting stages can be found in Section A.7. Partial experiment results are summarized in Table 2. Notably, the boosting performance highlighted in bold exhibits significant improvement compared to previously documented results [17, 23]. We see a pronounced enhancement with the blog dataset; before boosting, the Local prediction intervals are on average $42\%$ longer than those generated by CQR. After boosting, these intervals outperform the boosted CQR intervals by $32\%$. Using CQR as the baseline also yields substantial improvements, a decrease in averaged length exceeding $10\%$ in six out of the eleven datasets. The meps-21 dataset, in particular, shows an improvement of up to $18\%$ relative to the baseline. Results on the remaining datasets can be found in Tables A4 and A5. Figure 4 compares the conformalized prediction intervals derived from baseline Local and CQR scores with those obtained from the boosted scores. To effectively visualize the impact of boosting, we conduct a regression tree analysis on the training set to predict the label $Y$, setting the maximum number of tree nodes to four. This regression tree is then applied to the test set, allowing for a detailed comparison of the prediction intervals across each of the four distinct leaves.

# 7 Discussion

We introduced a post-training conformity score boosting scheme aiming to optimize for conditional coverage or length of the conformalized prediction interval. An intriguing avenue for future exploration involves simultaneously optimizing both length and conditional coverage, potentially trading

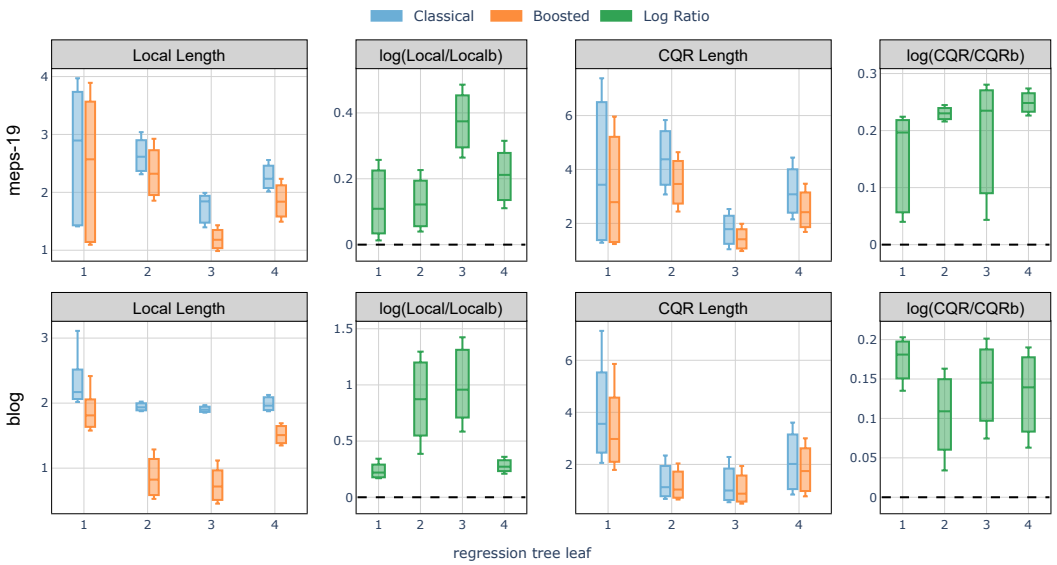



Figure 4: Comparison of test set average interval length evaluated on the meps-19 and blog datasets: classical Local and CQR conformal procedure versus the boosted procedures (abbreviated as 'Localb' and 'CQRb') compared in each of the 4 leaves of a regression tree trained on the training set to predict the label $Y$. A positive log ratio value between the regular and boosted interval lengths indicates improvement from boosting. The target miscoverage rate is set at $\alpha = 10\%$.

Table 2: Test set average interval length $\ell_L$ evaluated on various conformalized prediction intervals. The best result achieved for each dataset is highlighted in bold.

| Average Length, target miscoverage $\alpha = 10\%$ | | | | | | |
|---|---|---|---|---|---|---|
| Dataset | Method | | Improvement | Method | | Improvement |
| | Local | Boosted | | CQR | Boosted | |
| blog | 2.056 | **0.972** | -52.74% | 1.445 | 1.434 | -0.71% |
| facebook-1 | 1.896 | 1.383 | -27.03% | 1.198 | **1.072** | -10.47% |
| facebook-2 | 1.854 | 1.363 | -26.51% | 1.200 | **1.075** | -10.41% |
| meps-19 | 2.070 | **1.685** | -18.60% | 2.554 | 2.136 | -16.35% |
| meps-20 | 2.081 | **1.836** | -11.80% | 2.667 | 2.357 | -11.62% |
| meps-21 | 2.063 | **1.795** | -12.99% | 2.585 | 2.105 | -18.55% |

off these objectives by incorporating user-specified weights [29]. Additionally, we can readily adapt our procedure to meet various application-specific objectives. For instance, we can optimize for conditional coverage on predefined feature groups, a common task in enhancing fairness in distributing social resources across different demographic groups [27]. Similarly, we can modify our procedure to reduce the length of prediction intervals for predefined label groups, which can be seen as reallocating resources to decrease uncertainty for certain groups at the expense of higher uncertainty for other groups [26]. Candidate loss functions tailored to these objectives are detailed in Section A.1. Lastly, the primary emphasis of this paper centers on the design of the conformity score boosting scheme and formalizing the optimization of conditional coverage in mathematical terms, leaving room for computational optimization to enhance performance and runtime efficiency. In essence, the gradient boosting algorithm in our procedure can be replaced with any gradient-based machine learning model. Thus, another interesting future direction would be to explore whether alternative algorithms could enhance performance.

## Acknowlegements

E.C. was supported by the Office of Naval Research under Grant No. N00014-24-1-2305, the National Science Foundation under Grant No. DMS2032014, and the Simons Foundation under Award 814641. R.F.B. was supported by the Office of Naval Research via grant N00014-20-1-2337, and by the National Science Foundation via grant DMS-2023109.

## References


[1] Frank E. Harrell and C. E. Davis. "A New Distribution-Free Quantile Estimator". In: *Biometrika* 69.3 (1982), pp. 635–640. ISSN: 00063444. URL: http://www.jstor.org/stable/2335999.

[2] Stephen P Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

[3] "Classification with conformal predictors". In: *Algorithmic Learning in a Random World*. Boston, MA: Springer US, 2005, pp. 53–96. ISBN: 978-0-387-25061-8. DOI: 10.1007/0-387-25061-1_3. URL: https://doi.org/10.1007/0-387-25061-1_3.

[4] Vladimir Vovk, Alexander Gammerman, and Glenn Shafer. *Algorithmic learning in a random world*. Vol. 29. Springer, 2005.

[5] I-Cheng Yeh. *Concrete Compressive Strength*. UCI Machine Learning Repository. DOI: https://doi.org/10.24432/C5PK67. 2007.

[6] C.M. Achilles et al. *Tennessee's student teacher achievement ratio (STAR) project*. 2008.

[7] Michael Redmond. *Communities and Crime*. UCI Machine Learning Repository. DOI: https://doi.org/10.24432/C53W3X. 2009.

[8] Vladimir Vovk. "Conditional validity of inductive conformal predictors". In: *Asian conference on machine learning*. PMLR. 2012, pp. 475–490.

[9] Hadi Fanaee-T. *Bike Sharing Dataset*. UCI Machine Learning Repository. DOI: https://doi.org/10.24432/C5W894. 2013.

[10] Prashant Rana. *Physicochemical Properties of Protein Tertiary Structure*. UCI Machine Learning Repository. DOI: https://doi.org/10.24432/C5QW3H. 2013.

[11] Krisztian Buza. *BlogFeedback*. UCI Machine Learning Repository. DOI: https://doi.org/10.24432/C58S3F. 2014.

[12] Tianqi Chen and Carlos Guestrin. "Xgboost: A scalable tree boosting system". In: *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*. 2016, pp. 785–794.

[13] Kamaljot Singh. *Facebook Comment Volume Dataset*. UCI Machine Learning Repository. DOI: https://doi.org/10.24432/C5Q886. 2016.

[14] Andre Esteva et al. "Dermatologist-level classification of skin cancer with deep neural networks". In: *nature* 542.7639 (2017), pp. 115–118.

[15] Guolin Ke et al. "Lightgbm: A highly efficient gradient boosting decision tree". In: *Advances in neural information processing systems* 30 (2017).

[16] Jing Lei et al. "Distribution-Free Predictive Inference for Regression". In: *Journal of the American Statistical Association* 113.523 (2018), pp. 1094–1111. DOI: 10.1080/01621459.2017.1307116. eprint: https://doi.org/10.1080/01621459.2017.1307116. URL: https://doi.org/10.1080/01621459.2017.1307116.

[17] Yaniv Romano, Evan Patterson, and Emmanuel Candes. "Conformalized quantile regression". In: *Advances in neural information processing systems* 32 (2019).

[18] Yaniv Romano, Evan Patterson, and Emmanuel J. Candès. "Conformalized Quantile Regression". In: *Proceedings of the 33rd International Conference on Neural Information Processing Systems*. Red Hook, NY, USA: Curran Associates Inc., 2019.

[19] Mathieu Blondel et al. "Fast differentiable sorting and ranking". In: *International Conference on Machine Learning*. PMLR. 2020, pp. 950–959.

[20] Rina Foygel Barber et al. "The limits of distribution-free conditional predictive inference". In: *Information and Inference: A Journal of the IMA* 10.2 (Aug. 2020), pp. 455–482. ISSN: 2049-8772. DOI: 10.1093/imaiai/iaaa017. eprint: https://academic.oup.com/imaiai/article-pdf/10/2/455/38549621/iaaa017.pdf. URL: https://doi.org/10.1093/imaiai/iaaa017.

[21] Jerome H. Friedman. "Contrast trees and distribution boosting". In: *Proceedings of the National Academy of Sciences* 117.35 (2020), pp. 21175–21184. DOI: 10.1073/pnas.1921562117. eprint: https://www.pnas.org/doi/pdf/10.1073/pnas.1921562117. URL: https://www.pnas.org/doi/abs/10.1073/pnas.1921562117.

[22] Danijel Kivaranovic, Kory D. Johnson, and Hannes Leeb. *Adaptive, Distribution-Free Prediction Intervals for Deep Networks*. 2020. arXiv: 1905.10634 [stat.ML].

[23] Matteo Sesia and Emmanuel J Candès. "A comparison of some conformal quantile regression methods". In: *Stat* 9.1 (2020), e261.

[24] John Jumper et al. "Highly accurate protein structure prediction with AlphaFold". In: *Nature* 596.7873 (2021), pp. 583–589.

[25] Kuang-Ting Ko et al. "Structure of the malaria vaccine candidate Pfs48/45 and its recognition by transmission blocking antibodies". In: *Nature Communications* 13.1 (2022), p. 5603.

[26] David Stutz et al. "Learning Optimal Conformal Classifiers". In: *International Conference on Learning Representations*. 2022. URL: https://openreview.net/forum?id=t8O-4LKFVx.

[27] Isaac Gibbs, John J. Cherian, and Emmanuel J. Candès. *Conformal Prediction With Conditional Guarantees*. 2023. arXiv: 2305.12616 [stat.ME].

[28] Kexin Huang et al. *Uncertainty Quantification over Graph with Conformalized Graph Neural Networks*. 2023. arXiv: 2305.14535 [cs.LG].

[29] Lahav Dabah and Tom Tirer. *On Temperature Scaling and Conformal Prediction of Deep Classifiers*. 2024. arXiv: 2402.05806 [cs.LG]. URL: https://arxiv.org/abs/2402.05806.

[30] *Medical expenditure panel survey, panel 19*. URL: https://meps.ahrq.gov/mepsweb/data_stats/download_data_files_detail.jsp?cboPufNumber=HC-181.

[31] *Medical expenditure panel survey, panel 20*. URL: https://meps.ahrq.gov/mepsweb/data_stats/download_data_files_detail.jsp?cboPufNumber=HC-181.

[32] *Medical expenditure panel survey, panel 21*. URL: https://meps.ahrq.gov/mepsweb/data_stats/download_data_files_detail.jsp?cboPufNumber=HC-192.

# A Appendix

## A.1 Candidate loss functions for additional application-specific objectives

**Conditional coverage on predefined feature groups**: this task can be viewed as a specialized application within our broader strategy of boosting for conditional coverage, as detailed in Section 5. There, the primary challenge was to develop a loss function that accurately measures deviations from the target conditional coverage rate. We achieved this by using contrast trees to identify partitions in the feature space that maximize these deviations, effectively identifying subgroups in need of protection. This process is simplified when the partitions correspond to prespecified groups, allowing us to continue using the empirical maximum deviation as a candidate loss function.

Consider a dataset $\mathcal{D} = \{(X_i, Y_i)\}_{i=1}^n$ and a score function $E$. Denote by $C_n(\cdot)$ the conformalized prediction interval constructed from $E$. Let $G_1, \ldots, G_M$ be prespecified feature index groups. Within each set $\mathcal{D}_i = \{(X_j, Y_j)\}_{j \in G_i}$, compute the absolute deviation $d_i$ as

$$d_i\left(C_n(\cdot); \mathcal{D}_i\right) = \left| \frac{1}{|G_i|} \sum_{j \in G_i} \mathbb{1}(Y_j \in C_n(X_j)) - (1-\alpha) \right|. \tag{20}$$

The overall empirical maximum deviation is then defined as

$$\ell\left(E; \mathcal{D}\right) = \max_{1 \le i \le M} d_i\left(C_n(\cdot); \mathcal{D}_i\right). \tag{21}$$

**Interval length conditional on predefined label groups**: for a dataset $\mathcal{D} = \{(X_i, Y_i)\}_{i=1}^n$ and a score function $E$, let $\mathcal{Y}_1, \ldots, \mathcal{Y}_M$ be the prespecified label groups. A natural minimization objective for balancing uncertainty among these groups is defined as:

$$\ell(E; \mathcal{D}) = \sum_{i=1}^{M} w_i \frac{1}{\sum_{j=1}^n \mathbb{1}(Y_j \in \mathcal{Y}_i)} \sum_{j=1}^{n} \mathbb{1}(Y_j \in \mathcal{Y}_i) |C_n(X_j)|,$$

where $(w_1, \ldots, w_M)$ represents a set of user-specified weights.

## A.2 Proof of Proposition 5.1

Our proof relies on the following lemma.

**Lemma A.1** (Expressiveness)**.** *Given any sample pair $X$ and $Y$ with a continuous joint probability density distribution, and a prediction interval $[c_l(\cdot), c_u(\cdot)]$ with marginal coverage equal to $1-\alpha$, there exist specific function sets: ($\mu(\cdot)$,$\sigma(\cdot)$) for the Local type, and ($\mu_1(\cdot)$,$\mu_2(\cdot)$,$\tilde{\sigma}(\cdot)$) for the CQR type, such that asymptotically:*

1. *The conformalized prediction interval (7), derived using the generalized Local type conformity score $f_{\mu,\sigma}$, accurately recovers $[c_l(\cdot), c_u(\cdot)]$.*
2. *Similarly, the conformalized prediction interval (10), based on the generalized CQR type conformity score $E_{\mu_1,\mu_2,\tilde{\sigma}}$, also recovers $[c_l(\cdot), c_u(\cdot)]$.*

*Proof of Lemma A.1.* Recall that the generalized Local score (6) characterized by $(\mu, \sigma)$ takes the form

$$E_{\mu,\sigma}(x, y) = \frac{|y - \mu(x)|}{\sigma(x)}. \tag{22}$$

Asymptotically, the conformalized prediction interval is given by

$$\left[\mu(X) - Q_{1-\alpha}(E_{\mu,\sigma})\sigma(X), \mu(X) + Q_{1-\alpha}(E_{\mu,\sigma})\sigma(X)\right]. \tag{23}$$

Here, $Q_{1-\alpha}$ represents the population quantile. Set

$$\mu(x) = \frac{c_l(x) + c_u(x)}{2}, \quad \sigma(x) = \frac{c_u(x) - c_l(x)}{2}.$$

By assumption, we have

$$\mathbb{P}(Y \in [c_l(X), c_u(X)]) = 1 - \alpha.$$

With a simple change of variables, the above inequality is equivalent to

$$\mathbb{P}\left(\left|\frac{Y-\mu(X)}{\sigma(X)}\right| \leq 1\right) = 1-\alpha.$$

In other words, this is equivalent to

$$Q_{1-\alpha}(E_{\mu,\sigma}) = 1.$$

We have thus proved the result for the generalized Local type conformity score $E_{\mu,\sigma}$. In the same spirit, we can prove the result for the generalized CQR type conformity score $E_{\mu_1,\mu_2,\tilde{\sigma}}$ by taking

$$\mu_1 = c_l + \frac{c_u(x)-c_l(x)}{2}, \quad \mu_2 = c_u - \frac{c_u(x)-c_l(x)}{2}, \quad \tilde{\sigma} = \frac{c_u(x)-c_l(x)}{2}.$$

Recall that a generalized CQR score function (9) characterized by ($\mu_1$, $\mu_2$, $\sigma$) is defined as:

$$E_{\mu_1,\mu_2,\sigma}(x,y) = \max\left\{\mu_1(x)-y, y-\mu_2(x)\right\}/\sigma(x), \tag{24}$$

which leads to the asymptotic conformalized prediction intervals of the form

$$[\mu_1(X)-\sigma(X)Q_{1-\alpha}(E_{\mu_1,\mu_2,\sigma}), \mu_2(X)+\sigma(X)Q_{1-\alpha}(E_{\mu_1,\mu_2,\sigma})]. \tag{25}$$

Plugging in $\mu_1$, $\mu_2$, $\tilde{\sigma}$ defined above, we immediately have

$$\begin{aligned}
&\mathbb{P}(Y \in [c_l(X), c_u(X)]) = 1-\alpha \\
\iff\ &\mathbb{P}(c_l(X)-Y \leq 0, Y-c_u(X) \leq 0) = 1-\alpha \\
\iff\ &\mathbb{P}\left(\frac{c_l(X)-Y+\tilde{\sigma}(X)}{\tilde{\sigma}(X)} \leq 1, \frac{Y-c_u(X)+\tilde{\sigma}(X)}{\tilde{\sigma}(X)} \leq 1\right) = 1-\alpha \\
\iff\ &Q_{1-\alpha}(E_{\mu_1,\mu_2,\sigma}(x,y)) = 1.
\end{aligned}$$

□

*Proof of Proposition 5.1.* It suffices to take $c_l(x) = Q_{\alpha/2}(Y|X=x)$, $c_u(x) = Q_{1-\alpha/2}(Y|X=x)$ and apply Lemma A.1. □

### A.3 Boosting for length: theoretical guarantees

Similar to Proposition 5.1, we show in Proposition A.2 below that the generalized Local and CQR score families exhibit the necessary expressiveness to contain the oracle score, achieving optimal length while ensuring valid marginal coverage.

**Proposition A.2** (Asymptotic expressiveness)**.** *Under the assumptions of Proposition 5.1, for any target coverage rate $1-\alpha$, as $n \to \infty$, the following statements hold true:*

1. *There exists $(\mu^*, \sigma^*)$ such that the corresponding generalized Local score function (6) recovers the shortest oracle prediction interval (18).*
2. *There exists $(\mu_1^*, \mu_2^*, \sigma^*)$ such that the corresponding generalized CQR score function (9) recovers the shortest oracle prediction interval (18).*

*Proof of Proposition A.2.* It suffices to take $c_l(x) = Q_{l(\alpha)}(Y|X=x)$, $c_u(x) = Q_{u(\alpha)}(Y|X=x)$ and apply Lemma A.1, where $Q_{l(\alpha)}$ and $Q_{u(\alpha)}$ are the lower and upper limits of the High Density Region defined in (18). □

### A.4 CQR type conformity score boosting

A generalized CQR score function (9) is uniquely defined by a triple $(\mu_1(\cdot), \mu_2(\cdot), \sigma(\cdot))$. We will show how searching for a generalized CQR score can be reduced to searching for a Local generalized score. To begin with, we shall say that score functions are equivalent if they recover identical conformalized prediction intervals.

**Definition A.3.** *Let $\{X_i, Y_i\}_{i=1}^n$, $(X_{n+1}, Y_{n+1})$ be i.i.d. with continuous joint probability density distribution, and let $[n]$ be partitioned into a training set $I_1$ and a calibration set $I_2$. Consider two conformity score functions, $E_1$ and $E_2$, which produce conformalized prediction intervals $C_1(\cdot)$ and $C_2(\cdot)$, respectively. For any target coverage rate $1-\alpha$, $E_1$ and $E_2$ are equivalent if $C_1(\cdot) = C_2(\cdot)$ when marginal coverage rates $\mathbb{P}(Y_{n+1} \in C_1(X_{n+1}))$ and $\mathbb{P}(Y_{n+1} \in C_2(X_{n+1}))$ match.*

Building on this definition, we are now equipped to establish the following equivalences:

**Lemma A.4.** *Under the assumptions of Definition A.3, the following statements hold:*

1. *For the CQR-r score function defined in Section 2, there is an equivalent generalized Local score function characterized by a pair $(\mu(\cdot), \sigma(\cdot))$, where $\mu = (\mu_1 + \mu_2)/2$, $\sigma = (\mu_2 - \mu_1)/2$.*
2. *For any generalized Local score function characterized by the pair $(\mu(\cdot), \sigma(\cdot))$, there is an equivalent generalized CQR score function characterized by a triple $(\mu(\cdot), \mu(\cdot), \sigma(\cdot))$.*

The proof of the above Lemma is deferred to Section A.5. Leveraging these equivalences, we carry out the boosted conformal procedure as follows: first, we initialize a triple $\mu_1^{(0)} = \hat{q}_{\alpha/2}$, $\mu_2^{(0)} = \hat{q}_{1-\alpha/2}$, $\sigma_1^{(0)} = \hat{q}_{1-\alpha/2} - \hat{q}_{\alpha/2}$, which characterizes the CQR-r score function. Next, we find an equivalent generalized Local score function characterized by a pair $(\mu^{(0)}, \sigma^{(0)})$ chosen according to Lemma A.4. After $\tau$ boosting rounds, we obtain the boosted pair $(\mu^{(\tau)}, \sigma^{(\tau)})$ and the corresponding score function. Finally, we recover an equivalent generalized CQR score function

$$E^{(\tau)}(x, y) = \max\left\{\mu_1^{(\tau)}(x) - y, y - {\mu_2}^{(\tau)}(x)\right\} / \sigma_1^{(\tau)}(x),$$

characterized by the triple $({\mu_1}^{(\tau)}, {\mu_2}^{(\tau)}, {\sigma_1}^{(\tau)})$ chosen according to Lemma A.4.

### A.5 Proof of Lemma A.4

Recall that the generalized Local score (6) characterized by $(\mu, \sigma)$ takes the form

$$E_{\mu,\sigma}(x, y) = \frac{|y - \mu(x)|}{\sigma(x)}. \tag{26}$$

The conformalized prediction interval is given by

$$[\mu(X) - Q_{1-\alpha}(E_{\mu,\sigma}, I_2)\sigma(X), \mu(X) + Q_{1-\alpha}(E_{\mu,\sigma}, I_2)\sigma(X)]\,. \tag{27}$$

A generalized CQR score function (9) characterized by ($\mu_1$, $\mu_2$, $\sigma$) is defined as:

$$E_{\mu_1,\mu_2,\sigma}(x, y) = \max\{\mu_1(x) - y, y - \mu_2(x)\} / \sigma(x),$$

which leads to conformalized prediction intervals of the form

$$[\mu_1(X) - \sigma(X)Q_{1-\alpha}(E_{\mu_1,\mu_2,\sigma}, I_2), \mu_2(X) + \sigma(X)Q_{1-\alpha}(E_{\mu_1,\mu_2,\sigma}, I_2)].$$

1. Plugging in the triple $\mu_1(x) = \hat{q}_{\alpha/2}$, $\mu_2(x) = \hat{q}_{1-\alpha/2}$, $\sigma_1(x) = \hat{q}_{1-\alpha/2} - \hat{q}_{\alpha/2}$, which characterize the CQR-r score function, we have the conformalized prediction interval
$$[\mu_1(X) - \sigma_1(X)Q_{1-\alpha}(E_{\mu_1,\mu_2,\sigma_1}, I_2), \mu_2(X) + \sigma_1(X)Q_{1-\alpha}(E_{\mu_1,\mu_2,\sigma_1}, I_2)].$$
Set
$$\mu(X) = \frac{\mu_1(X) + \mu_2(X)}{2}, \sigma(X) = \frac{\mu_2(X) - \mu_1(X)}{2},$$
then the generalized Local conformity score $E_{\mu,\sigma}(x, y) = |y - \mu(x)|/\sigma(x)$ recovers conformalized prediction intervals of the form
$$\begin{aligned}
&[\mu(X) - \sigma(X)Q(E_{\mu,\sigma}, I_2), \mu(X) + \sigma(X)Q(E_{\mu,\sigma}, I_2)] \\
&= \left[\frac{\mu_1(X) + \mu_2(X)}{2} - \frac{\mu_2(X) - \mu_1(X)}{2}Q(E_{\mu,\sigma}, I_2),\right. \\
&\qquad\qquad \left.\frac{\mu_1(X) + \mu_2(X)}{2} + \frac{\mu_2(X) - \mu_1(X)}{2}Q(E_{\mu,\sigma}, I_2)\right] \\
&= \left[\mu_2(X) - (\mu_2(X) - \mu_1(X))\frac{Q(E_{\mu,\sigma}, I_2) - 1}{2},\right. \\
&\qquad\qquad \left.\mu_1(X) + (\mu_2(X) - \mu_1(X))\frac{Q(E_{\mu,\sigma}, I_2) - 1}{2}\right] \\
&= \left[\mu_1(X) - \sigma_1(X)\frac{Q(E_{\mu,\sigma}, I_2) - 1}{2}, \mu_2(X) + \sigma_1(X)\frac{Q(E_{\mu,\sigma}, I_2) - 1}{2}\right].
\end{aligned}$$
From the monotonicity of the interval lengths with respect to the empirical quantiles, we have that the two score functions are equivalent by Definition A.3.

2. Let a generalized Local score function be $E_{\mu,\sigma}(x, y) = |y - \mu(x)|/\sigma(x)$. Then it suffices to observe that

$$|y - \mu(x)| = \max\{y - \mu(z), \mu(x) - y\}.$$

### A.6 Additional information on real datasets

In Table A1, we provide the predicted label, dimensions, and source for each dataset. Data cleaning and preprocessing are in accordance with the methods described by Romano et al. [17].

Table A1: Datasets for our empirical analyses, with the predicted label, number of samples ($n$), and features ($d$).

| Name | Label | $n$ | $d$ | Source |
|---|---|---|---|---|
| bike | bike rental counts | 10886 | 18 | [9] |
| bio | deviation of predicted from native protein structure | 45730 | 9 | [10] |
| blog | number of comments in the next 24 hours | 52397 | 280 | [11] |
| community | crime rate per community | 1994 | 100 | [7] |
| concrete | concrete compressive strength | 1030 | 8 | [5] |
| facebook-1 | Facebook comment volume | 40948 | 53 | [5] |
| facebook-2 | Facebook comment volume | 81311 | 53 | [13] |
| meps-19 | utilization of medical services | 15785 | 139 | [30] |
| meps-20 | utilization of medical services | 17541 | 139 | [31] |
| meps-21 | utilization of medical services | 15656 | 139 | [32] |
| star | total student test scores up to the third grade | 2161 | 39 | [6] |

All datasets, except for the meps and star data sets, are licensed under CC-BY 4.0. The Medical Expenditure Panel Survey (meps) data is subject to copyright and usage rules. The licensing status of the star dataset could not be determined.

### A.7 Experimental Setup

In each dataset, we randomly hold out $20\%$ as test data. The remaining data is divided into a training set and a calibration set, each taking up a proportion of $\gamma$ and $1 - \gamma$. We explore training ratios $\gamma$ ranging from $10\%$ to $90\%$. Results corresponding to the optimal value of the hyperparameter $\gamma$ are recorded in Table 1, following the practice of Sesia er al. [23].

In the training stage, we employ the random forest regressor from Python's scikit-learn package to learn the baseline Local score function. The hyperparameters are the package defaults, except for the total number of trees, which we set to 1000, and the minimum number of samples required at a leaf node, which we set to 40, as recommended by Romano et al. [17]. For the baseline CQR score function, we adopt a black-box neural network quantile regressor with three fully connected layers and ReLU non-linearities, following the practice of Sesia et al. [23]. In the boosting stage, we set the hyper-parameters $\tau_1$, $\tau_2$ in the approximated loss (17) to 50. The approximated loss is then passed to the Gradient Boosting Machine from Python's XGBoost package along with a base conformity score. We set the maximum tree depth to 1 to avoid overfitting and perform cross-validation for the number of boosting rounds, as outlined in Section 3. All other hyperparameters are set to package defaults.

All experiments were conducted on a dual-socket AMD EPYC 7502 32-Core Processor system, utilizing 8 of its 128 CPUs each time. The runtime for each dataset and random seed varies by dataset size, ranging from 10 minutes to 5 hours.

### A.8 Additional results and error bars

In Tables A2 to A5, we present additional results on marginal coverage, maximum conditional coverage deviation ($\ell_M$), and average interval length ($\ell_P$) for each real dataset (including those not reported in Tables 1 and 2), both before and after boosting. Notably, in each case, boosting is applied to optimize either conditional coverage or average interval length. As a result, the non-targeted characteristic may or may not improve after boosting.

Table A2: Additional information on conformalized intervals obtained before and after boosting for conditional coverage with the Local conformity score as baseline. The target miscoverage rate is set to $\alpha = 10\%$.

| Dataset | $\ell_L$ | | $\ell_M(\%)$ | | Marginal Cov.(%) | |
|---|---|---|---|---|---|---|
| | Local | Boosted | Local | Boosted | Local | Boosted |
| bike | 1.775 | 2.201 | 10.979 | 5.638 | 89.927 | 89.646 |
| bio | 1.602 | 1.614 | 5.303 | 4.862 | 90.024 | 90.093 |
| blog | 2.080 | 3.403 | 51.353 | 48.591 | 89.995 | 90.040 |
| community | 1.824 | 10.759 | 25.755 | 13.466 | 90.376 | 89.549 |
| concrete | 1.058 | 1.062 | 10.740 | 8.763 | 90.583 | 91.359 |
| facebook-1 | 1.896 | 4.790 | 26.020 | 25.917 | 90.201 | 90.056 |
| facebook-2 | 1.881 | 2.273 | 42.807 | 42.437 | 89.966 | 89.989 |
| meps-19 | 2.074 | 2.926 | 15.357 | 5.656 | 90.120 | 90.497 |
| meps-20 | 2.102 | 2.778 | 16.939 | 6.998 | 89.963 | 90.031 |
| meps-21 | 2.069 | 2.537 | 17.627 | 7.832 | 90.064 | 90.054 |
| star | 0.189 | 0.179 | 9.658 | 9.348 | 90.808 | 90.831 |

Table A3: Additional information on conformalized intervals obtained before and after boosting for conditional coverage with the CQR conformity score as baseline. The target miscoverage rate is set to $\alpha = 10\%$.

| Dataset | $\ell_L$ | | $\ell_M(\%)$ | | Marginal Cov.(%) | |
|---|---|---|---|---|---|---|
| | CQR | Boosted | CQR | Boosted | CQR | Boosted |
| bike | 0.555 | 0.540 | 4.934 | 4.925 | 90.073 | 90.184 |
| bio | 1.518 | 1.515 | 5.069 | 4.700 | 89.841 | 89.853 |
| blog | 1.761 | 1.766 | 27.760 | 26.836 | 90.222 | 90.244 |
| community | 1.718 | 1.740 | 12.688 | 12.105 | 90.340 | 90.194 |
| concrete | 0.484 | 0.489 | 9.039 | 8.265 | 90.451 | 90.652 |
| facebook-1 | 1.374 | 1.371 | 13.407 | 13.255 | 90.465 | 90.247 |
| facebook-2 | 1.465 | 1.409 | 18.257 | 18.002 | 89.763 | 90.001 |
| meps-19 | 2.784 | 2.784 | 5.507 | 5.507 | 90.257 | 90.257 |
| meps-20 | 2.769 | 2.743 | 7.614 | 7.184 | 89.991 | 90.006 |
| meps-21 | 2.834 | 2.815 | 8.16 | 8.067 | 90.169 | 90.067 |
| star | 0.199 | 0.209 | 9.728 | 9.630 | 91.085 | 91.339 |

Table A4: Additional information on conformalized intervals obtained before and after boosting for length with the Local conformity score as baseline. The target miscoverage rate is set to $\alpha = 10\%$.

| Dataset | $\ell_L$ | | $\ell_M(\%)$ | | Marginal Cov.(%) | |
|---|---|---|---|---|---|---|
| | Local | Boosted | Local | Boosted | Local | Boosted |
| bike | 1.775 | 1.360 | 22.590 | 19.616 | 89.927 | 89.862 |
| bio | 1.562 | 1.514 | 5.995 | 5.791 | 89.937 | 89.962 |
| blog | 2.056 | 0.972 | 52.440 | 54.858 | 89.988 | 89.978 |
| community | 1.728 | 1.678 | 26.066 | 24.822 | 89.499 | 89.323 |
| concrete | 1.010 | 0.698 | 11.029 | 10.518 | 90.631 | 90.728 |
| facebook-1 | 1.896 | 1.384 | 26.020 | 34.259 | 90.201 | 89.944 |
| facebook-2 | 1.854 | 1.363 | 42.624 | 50.697 | 90.020 | 89.972 |
| meps-19 | 2.070 | 1.685 | 18.626 | 14.623 | 90.054 | 90.070 |
| meps-20 | 2.081 | 1.836 | 17.897 | 14.643 | 89.869 | 89.849 |
| meps-21 | 2.063 | 1.795 | 18.795 | 13.324 | 89.914 | 89.920 |
| star | 0.179 | 0.179 | 9.976 | 9.407 | 90.901 | 90.577 |

Table A5: Additional information on conformalized intervals obtained before and after boosting for length with the CQR conformity score as baseline. The target miscoverage rate is set to $\alpha = 10\%$.

| Dataset | $\ell_L$ | | $\ell_M(\%)$ | | Marginal Cov.(%) | |
|---|---|---|---|---|---|---|
| | CQR | Boosted | CQR | Boosted | CQR | Boosted |
| bike | 0.553 | 0.489 | 9.530 | 10.092 | 90.041 | 90.418 |
| bio | 1.516 | 1.468 | 5.408 | 5.265 | 89.670 | 89.880 |
| blog | 1.445 | 1.434 | 29.875 | 34.011 | 90.149 | 90.260 |
| community | 1.693 | 1.699 | 14.006 | 13.536 | 89.499 | 89.699 |
| concrete | 0.391 | 0.393 | 10.342 | 10.700 | 88.932 | 89.223 |
| facebook-1 | 1.198 | 1.073 | 20.132 | 30.901 | 89.937 | 90.013 |
| facebook-2 | 1.200 | 1.075 | 29.233 | 34.475 | 90.035 | 90.010 |
| meps-19 | 2.554 | 2.136 | 10.357 | 11.285 | 90.228 | 90.399 |
| meps-20 | 2.667 | 2.357 | 10.826 | 10.720 | 89.875 | 89.838 |
| meps-21 | 2.585 | 2.105 | 10.968 | 10.565 | 89.946 | 89.863 |
| star | 0.195 | 0.194 | 9.982 | 10.142 | 91.455 | 91.432 |

We have previously reported the evaluated losses $\ell_M$ and $\ell_L$ for each dataset, averaged over ten random seeds. Tables A6 and A7 below detail the distribution of these evaluations, providing the mean, $10\%$ quantile, and $90\%$ quantile for the test set deviations in conditional coverage ($\ell_M$) and average interval length ($\ell_L$). These statistics are derived from 110 test set evaluations across 11 datasets and 10 random training-test splits. We opt to report empirical quantiles instead of standard deviations due to the asymmetric and non-Gaussian nature of the data.

Table A6: Distribution of the test set conditional coverage deviation $\ell_M$ evaluated on various conformalized prediction intervals across 11 datasets and 10 random training-test splits.

| Max. Conditional Coverage Deviation (%), target miscoverage $\alpha = 10\%$ | | | | |
|---|---|---|---|---|
| Statistics | Method | | Method | |
| | Local | Boosted | CQR | Boosted |
| mean | 21.140% | 16.319% | 11.106% | 10.771% |
| 10% quantile | 7.267% | 4.604% | 4.890% | 4.910% |
| 90% quantile | 47.832% | 44.712% | 22.585% | 18.697% |

Table A7: Distribution of the test set average interval length $\ell_L$ evaluated on various conformalized prediction intervals across 11 datasets and 10 random training-test splits.

| Average Length, target miscoverage $\alpha = 10\%$ | | | | |
|---|---|---|---|---|
| Statistics | Method | | Method | |
| | Local | Boosted | CQR | Boosted |
| mean | 1.677 | 1.317 | 1.483 | 1.319 |
| 10% quantile | 0.950 | 0.513 | 0.346 | 0.351 |
| 90% quantile | 2.082 | 1.829 | 2.655 | 2.210 |

### A.9 Experiments under different miscoverage rates

In Tables A8 and A9, we illustrate the performance of boosting for length with the Local conformity score as baseline with miscoverage rates set to $5\%$ and $20\%$, respectively.

Table A8: Additional information on conformalized intervals obtained before and after boosting for length with the Local conformity score as baseline. The target miscoverage rate is set to $\alpha = 5\%$.

| Dataset | $\ell_L$ | | | $\ell_M(\%)$ | | Marginal Cov.(%) | |
|---|---|---|---|---|---|---|---|
| | Local | Boosted | Improvement | Local | Boosted | Local | Boosted |
| bike | 2.523 | 1.828 | -27.54% | 11.553 | 11.626 | 94.927 | 94.972 |
| bio | 1.881 | 1.801 | -4.25% | 5.013 | 4.694 | 94.857 | 94.802 |
| blog | 4.217 | 1.923 | -54.39% | 37.976 | 34.505 | 95.027 | 95.048 |
| community | 2.578 | 2.217 | -13.99% | 16.406 | 14.554 | 95.414 | 94.812 |
| concrete | 1.129 | 0.838 | -25.80% | 8.55 | 8.086 | 94.709 | 94.515 |
| facebook-1 | 2.667 | 2.262 | -15.18% | 20.85 | 20.052 | 95.104 | 95.049 |
| facebook-2 | 2.441 | 2.092 | -14.30% | 41.645 | 39.943 | 94.941 | 95.016 |
| meps-19 | 3.618 | 2.825 | -21.92% | 10.515 | 9.178 | 95.005 | 95.008 |
| meps-20 | 3.951 | 3.049 | -22.84% | 11.043 | 9.693 | 94.907 | 94.922 |
| meps-21 | 4.030 | 3.033 | -24.74% | 8.918 | 8.910 | 95.061 | 95.026 |
| star | 0.207 | 0.207 | -0.11% | 6.57 | 6.906 | 95.358 | 95.289 |

Table A9: Additional information on conformalized intervals obtained before and after boosting for length with the Local conformity score as baseline. The target miscoverage rate is set to $\alpha = 20\%$.

| Dataset | $\ell_L$ | | | $\ell_M(\%)$ | | Marginal Cov.(%) | |
|---|---|---|---|---|---|---|---|
| | Local | Boosted | Improvement | Local | Boosted | Local | Boosted |
| bike | 1.312 | 0.983 | -25.07% | 24.649 | 28.561 | 79.403 | 80.152 |
| bio | 1.248 | 1.213 | -2.80% | 10.838 | 10.540 | 79.729 | 79.594 |
| blog | 1.904 | 0.513 | -73.07% | 45.813 | 64.360 | 79.756 | 79.920 |
| community | 1.337 | 1.234 | -7.71% | 25.047 | 28.831 | 79.674 | 79.674 |
| concrete | 0.833 | 0.549 | -34.10% | 13.831 | 20.177 | 80.728 | 80.825 |
| facebook-1 | 1.624 | 0.747 | -54.02% | 29.384 | 47.994 | 80.190 | 79.731 |
| facebook-2 | 1.580 | 0.749 | -52.60% | 36.488 | 62.070 | 79.890 | 79.930 |
| meps-19 | 1.821 | 1.020 | -44.03% | 20.561 | 24.763 | 80.326 | 80.013 |
| meps-20 | 1.843 | 1.095 | -40.60% | 19.137 | 24.194 | 79.684 | 79.823 |
| meps-21 | 1.831 | 1.064 | -41.86% | 18.227 | 23.172 | 80.674 | 80.057 |
| star | 0.142 | 0.141 | -0.71% | 15.420 | 14.517 | 80.647 | 80.370 |

### A.10 Training a gradient boosting algorithm with our custom loss functions

Our proposed boosted conformal procedure serves as a post-training step designed to refine the conformity score $E(\cdot, \cdot; f)$ obtained during model training. This procedure can leverage pre-trained models when available. In the absence of pre-trained models, we can alternatively train a gradient boosting algorithm directly using our custom loss functions. For example, in the context of the local score, we may initialize with $\mu^{(0)} = 0$, $\sigma^{(0)} = 1$ and then apply our boosted conformal procedure. As discussed in Section 7, this approach is flexible enough to replace gradient boosting with any gradient-based algorithm, such as neural networks, trained under our custom loss functions. This framework aligns with that of [26], which introduces a neural network trained with a custom loss function to minimize the average prediction set size in classification tasks.

In Figure A1, we compare the performance of the two approaches optimizing for average interval length, searching within the generalized Local score family $\mathcal{F}$ defined in (6). The primary distinction between the two procedures lies in the initialization: the first approach employs $\mu^{(0)} = 0$, $\sigma^{(0)} = 1$, while the second derives $\mu^{(0)}$ and $\sigma^{(0)}$ from a trained random forests model. We run the experiments on the meps-19 dataset and compare the performance across different splits of the training and calibration data. In this context, the percentage of training data refers to the proportion of training data within the combined training and calibration datasets. Our results indicate that cross-validation selects a greater number of boosting iterations when we directly train the gradient boosting algorithm, resulting in longer runtime. However, the average interval length and maximum conditional coverage deviation after boosting are notably smaller for the boosted conformal procedure we introduced in this paper.

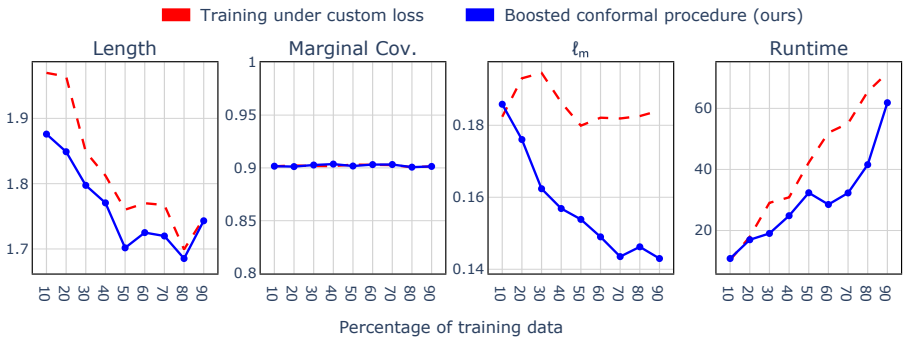



Figure A1: Comparison of boosted interval length, marginal coverage, maximum conditional coverage deviation ($\ell_M$), and runtime between direct training of a gradient-based algorithm (red) and boosting on a pre-trained conformity score (blue).

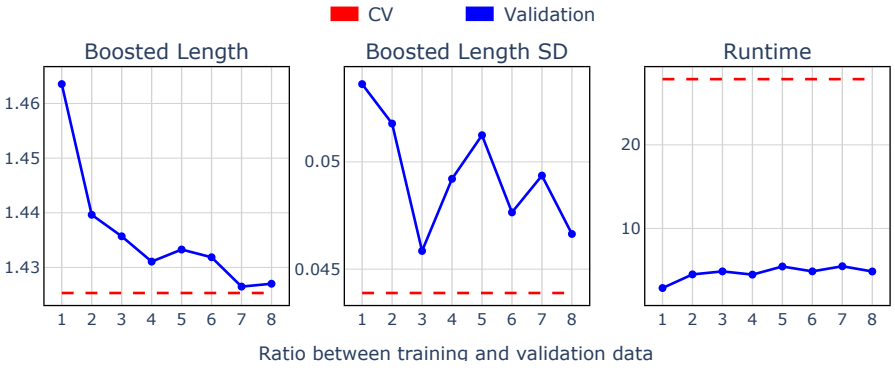



Figure A2: Comparison of the average boosted length, standard deviation of boosted length, and average runtime of the boosting procedure when selecting the optimal number of boosting rounds using 5-fold cross-validation versus a hold-out validation set of varying sizes. Experiments are conducted on the bike dataset, with a target miscoverage rate of $\alpha = 10\%$.

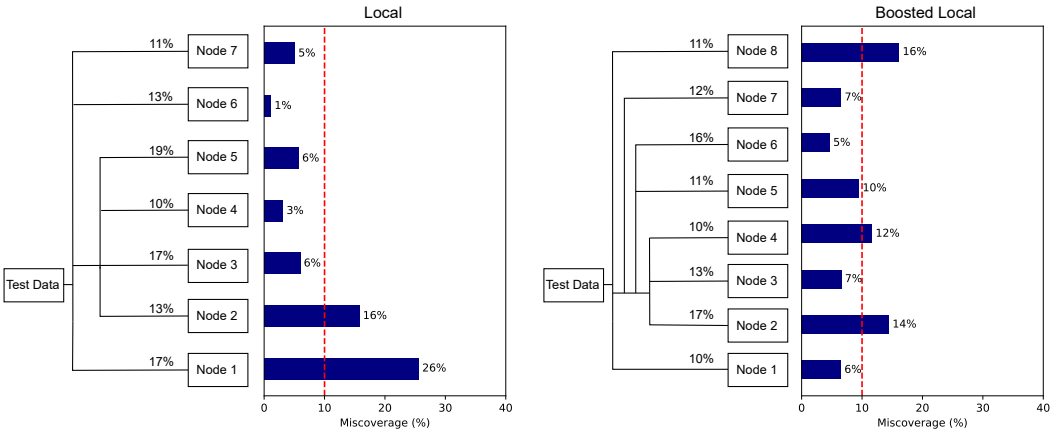
### A.11 Selecting optimal boosting rounds via hold-out validation set

In our boosted conformal procedure, we use cross-validation on the training set to determine the optimal number of boosting rounds, a process that can be time-consuming. An alternative approach is to hold out a fraction of the training set for validation. While more computationally efficient, this method introduces a trade-off: a smaller validation set can lead to greater variability in prediction intervals and model performance, whereas a larger validation set may reduce the effective training set size, potentially limiting the model's performance. To explore this trade-off, we conduct experiments on the bike dataset, optimizing for prediction interval length. We compare performance across two settings: 5-fold cross-validation and a hold-out validation set, with the training-to-validation set ratio ranging from 1:1 to 8:1. For each setting, we run 100 experiments, recording the average boosted length, the standard deviation of boosted lengths, and the average runtime. The results are shown in Figure A2.

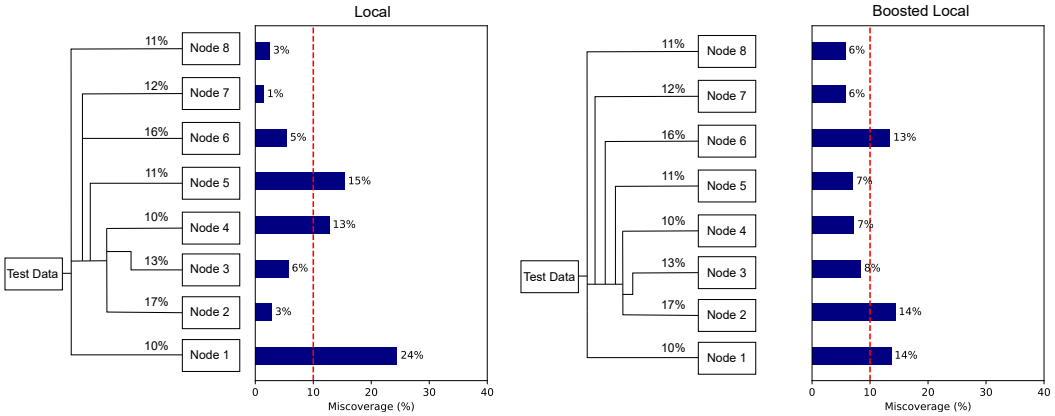
### A.12 Additional figures on individual datasets

In this section, we present a series of supplementary figures. First, we showcase the improvements in conditional coverage achieved through the boosted procedure for each benchmark dataset. Figure A3 details results for datasets meps-20 and meps-21. Figure A4 details results for datasets community, bike, and concrete.

Next, we illustrate enhanced interval lengths. Figure A5 details results for datasets meps-20, meps-21, and bike. Figure A6 details results for datasets facebook-1, facebook-2, and concrete. Finally, we demonstrate in Figure A7 how cross-validating the number of boosting rounds effectively prevents the gradient boosting algorithm from overfitting.

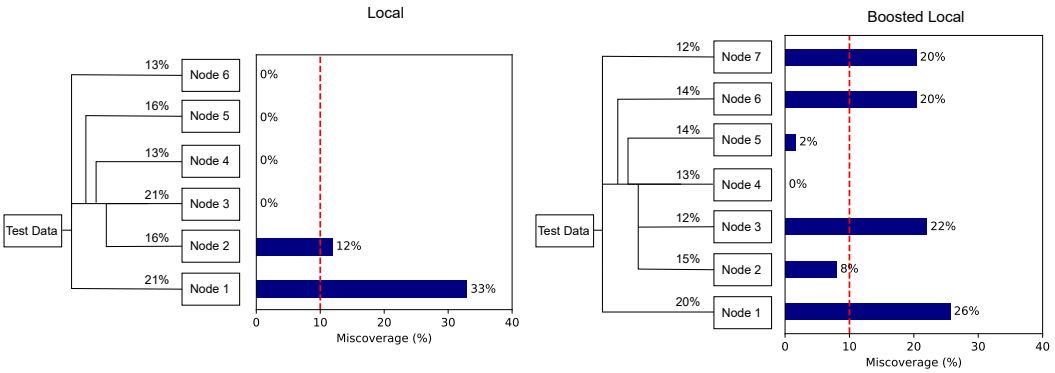



(a) meps-20 dataset.

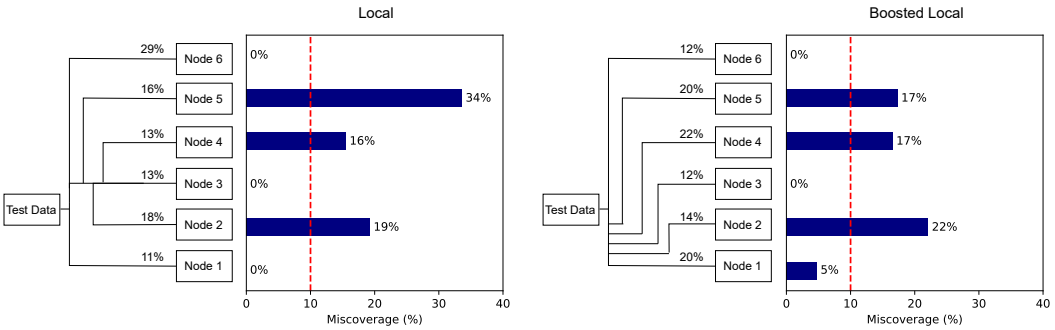



(b) meps-21 dataset.

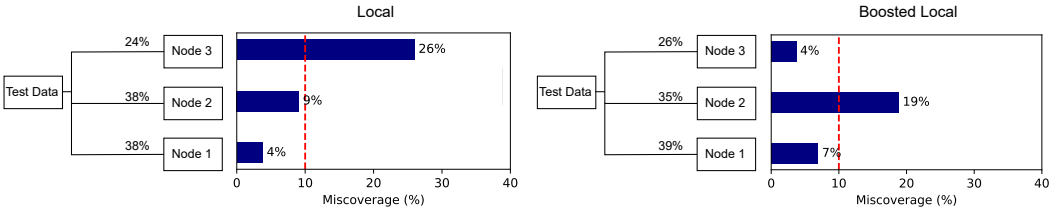
Figure A3: See the caption of Figure 3 for details.




(a) community dataset.

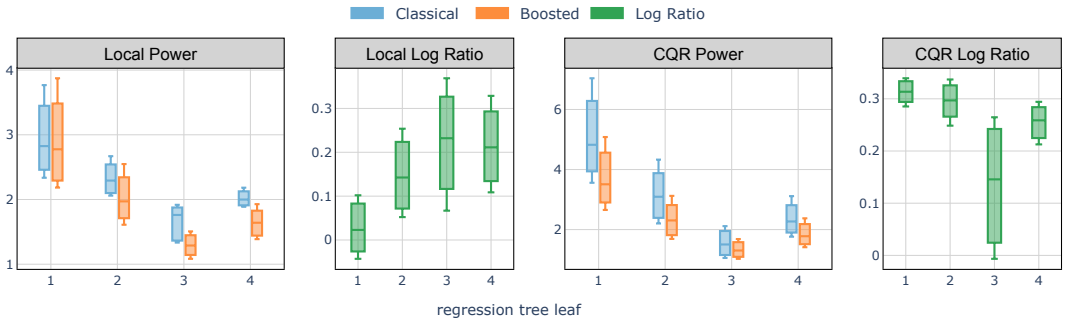



(b) bike dataset.

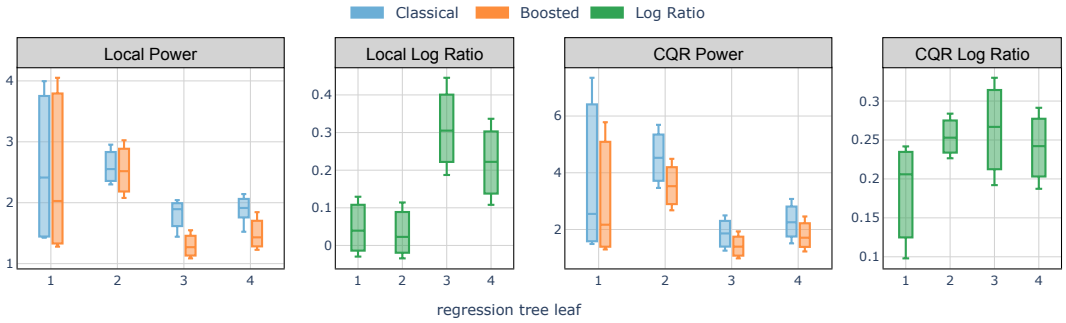



(c) concrete dataset.

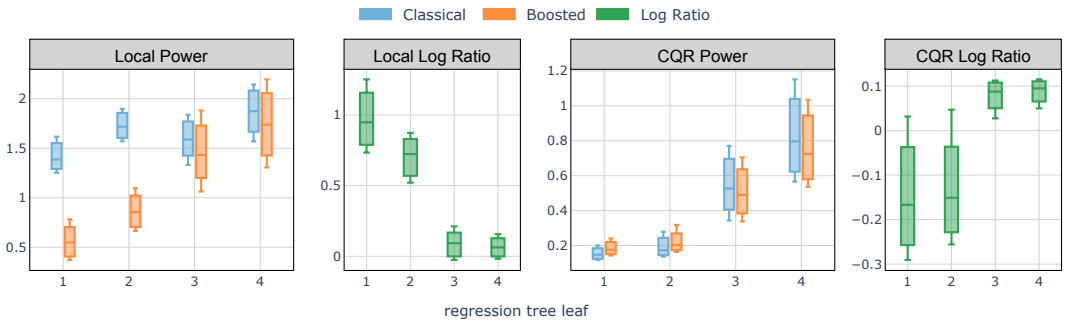
Figure A4: See the caption of Figure 3 for details.




(a) meps-20 dateset.

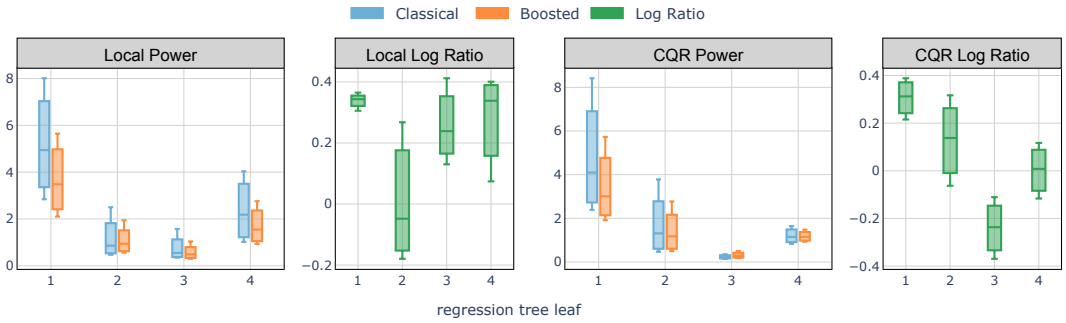



(b) meps-21 dateset.

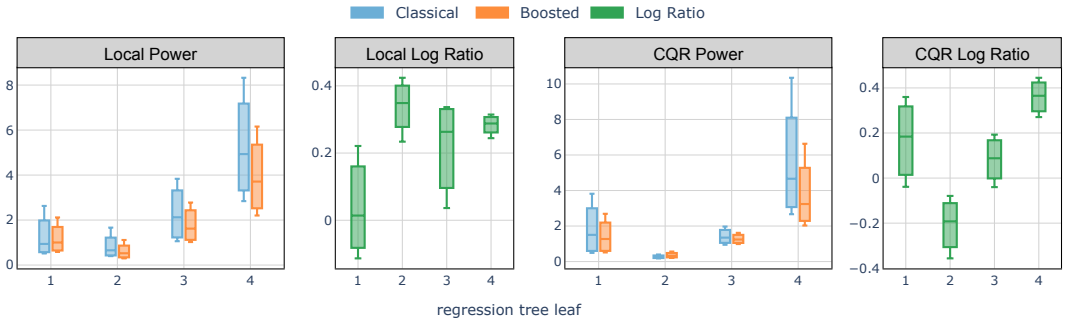



(c) bike dateset.

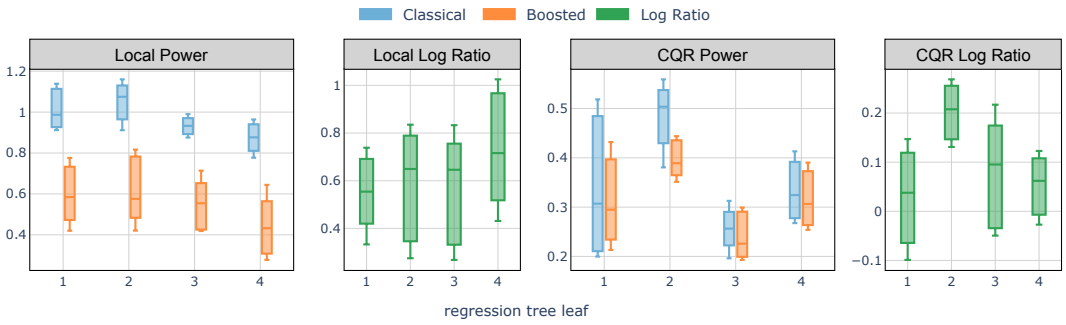
Figure A5: See the caption of Figure 4 for details.




(a) facebook-1 dateset.

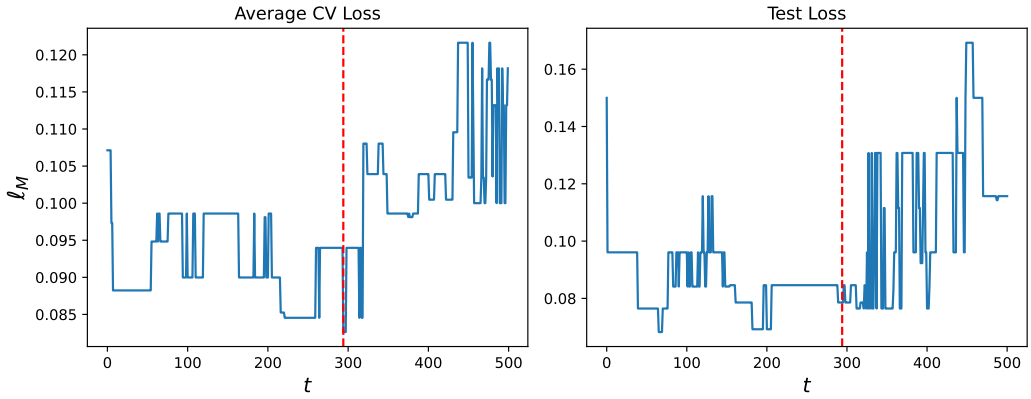



(b) facebook-2 dateset.

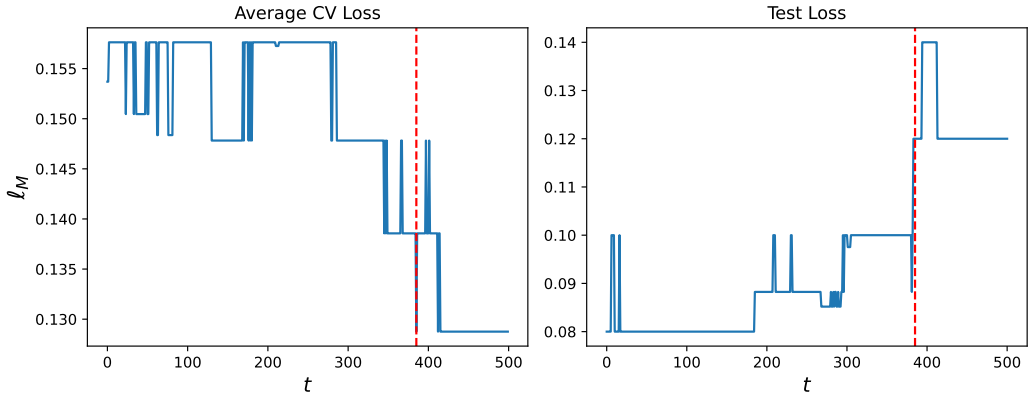



(c) concrete dateset.

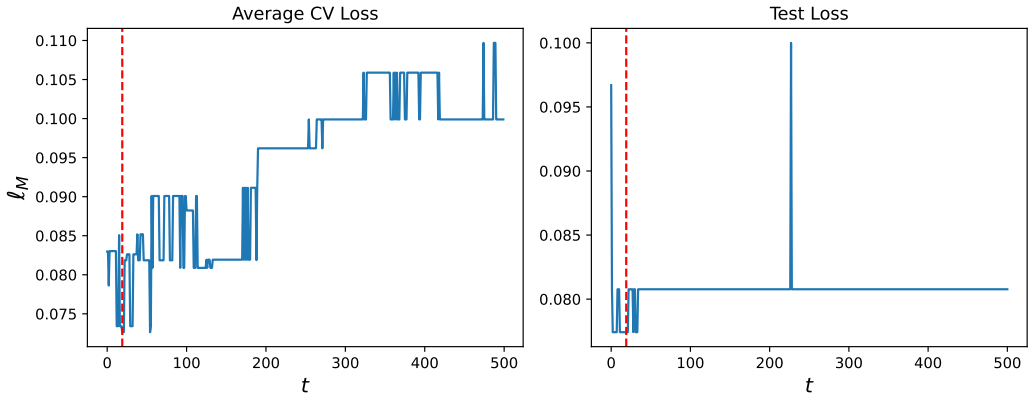
Figure A6: See the caption of Figure 4 for details.




(a) seed 9.




(b) seed 8.




(c) seed 7.

Figure A7: Empirical maximum deviation $\ell_M$ across $T = 500$ boosting rounds evaluated on dataset concrete under random seeds 7, 8, 9, train-calibration ratio $60\%$ : The left panel illustrates the cross-validated loss, computed as the average across $k = 3$ sub-calibration folds. The right panel displays the test loss. The optimal number of boosting rounds $\tau$, determined through cross-validation as specified in (12), is highlighted in red.

## NeurIPS Paper Checklist

The checklist is designed to encourage best practices for responsible machine learning research, addressing issues of reproducibility, transparency, research ethics, and societal impact. Do not remove the checklist: **The papers not including the checklist will be desk rejected.** The checklist should follow the references and precede the (optional) supplemental material. The checklist does NOT count towards the page limit.

Please read the checklist guidelines carefully for information on how to answer these questions. For each question in the checklist:

- You should answer [Yes] , [No] , or [NA] .
- [NA] means either that the question is Not Applicable for that particular paper or the relevant information is Not Available.
- Please provide a short (1–2 sentence) justification right after your answer (even for NA).

**The checklist answers are an integral part of your paper submission.** They are visible to the reviewers, area chairs, senior area chairs, and ethics reviewers. You will be asked to also include it (after eventual revisions) with the final version of your paper, and its final version will be published with the paper.

The reviewers of your paper will be asked to use the checklist as one of the factors in their evaluation. While "[Yes] " is generally preferable to "[No] ", it is perfectly acceptable to answer "[No] " provided a proper justification is given (e.g., "error bars are not reported because it would be too computationally expensive" or "we were unable to find the license for the dataset we used"). In general, answering "[No] " or "[NA] " is not grounds for rejection. While the questions are phrased in a binary way, we acknowledge that the true answer is often more nuanced, so please just use your best judgment and write a justification to elaborate. All supporting evidence can appear either in the main paper or the supplemental material, provided in appendix. If you answer [Yes] to a question, in the justification please point to the section(s) where related material for the question can be found.

IMPORTANT, please:

- **Delete this instruction block, but keep the section heading "NeurIPS paper checklist"**,
- **Keep the checklist subsection headings, questions/answers and guidelines below.**
- **Do not modify the questions and only use the provided macros for your answers**.

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification:

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Assumptions necessary for establishing the theoretical guarantees are detailed in each result presented in the paper. Additional limitations are discussed in Section 7.

Guidelines:

- The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: See Sections A.4, A.2 and A.3.

   Guidelines:

   - The answer NA means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: See Sections 5.2.2 and A.7.

   Guidelines:

   - The answer NA means that the paper does not include experiments.

- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [No]

   Justification: Real data sets for our empirical analyses are publicly accessible and described in Section A.6. Currently, the code is accessible upon request. Although the code has not been released yet, we have provided detailed documentation for reproducibility in Sections 5.2.2 and A.7.

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.

- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: See Section A.7.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: See Section A.8.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
   - If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: See Section A.7.

   Guidelines:

   - The answer NA means that the paper does not include experiments.

- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification:

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: See Introduction.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: The paper poses no such risks.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: See Section A.7.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: See Sections 5.2.2, A.7, and A.6.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [NA]

Justification: The paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [NA]

Justification: The paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.