

# Conditional Coverage Diagnostics for Conformal Prediction

Sacha Braun<sup>1,2</sup>, David Holzmüller<sup>3</sup>, Michael I. Jordan<sup>1,4</sup>, and Francis Bach<sup>1,2</sup>

<sup>1</sup>Sierra team, Inria Paris, France

{sacha.braun, francis.bach}@inria.fr

<sup>2</sup>Ecole Normale Supérieure, PSL Research University, Paris

<sup>3</sup>Soda team, Inria Paris-Saclay, France

david.holzmuller@inria.fr

<sup>4</sup>Departments of EECS and Statistics, UC Berkeley, USA

jordan@cs.berkeley.edu

## Abstract

Evaluating conditional coverage remains one of the most persistent challenges in assessing the reliability of predictive systems. Although conformal methods can give guarantees on marginal coverage, no method can guarantee to produce sets with correct conditional coverage, leaving practitioners without a clear way to interpret local deviations. To overcome sample-inefficiency and overfitting issues of existing metrics, we cast conditional coverage estimation as a classification problem. Conditional coverage is violated if and only if any classifier can achieve lower risk than the target coverage. Through the choice of a (proper) loss function, the resulting risk difference gives a conservative estimate of natural miscoverage measures such as L1 and L2 distance, and can even separate the effects of over- and under-coverage, as well as handle non-constant target coverages. We call the resulting family of metrics *excess risk of the target coverage* (ERT). We show experimentally that the use of modern classifiers provides much higher statistical power than simple classifiers underlying established metrics like CovGap. Additionally, we use our metric to benchmark different conformal prediction methods. Finally, we release an open-source package for ERT as well as previous conditional coverage metrics. Together, these contributions provide a new lens for understanding, diagnosing, and improving the conditional reliability of predictive systems.

## 1 Introduction

Uncertainty quantification is central to decision-making across science, engineering, and policy. In many applications, the goal is not a single point prediction but a set of plausible outcomes with a desired confidence level. This is formalized by a predictive set rule  $C(\cdot)$ , which outputs a region expected to contain the true outcome with a desired probability. Such predictive sets capture data noise, model imperfections, and variability, supporting safer decisions and clearer communication of model confidence.

Conformal prediction (CP) offers a general framework for constructing prediction sets with finite-sample coverage guarantees (Vovk et al., 2005; Shafer and Vovk, 2008). Its only requirement is that the available data samples are exchangeable: a condition even weaker than the standard independent and identically distributed (i.i.d.) assumption. In recent years, CP has rapidly emerged as a go-to tool for adding rigorous, model-agnostic uncertainty estimates to modern black-box predictors (Angelopoulos and Bates, 2023). This makes it especially appealing for scientific and industrial applications where formal guarantees on model predictions are essential.

CP’s simplicity hides an important drawback: it only guarantees *marginal coverage*, which means that the constructed prediction set contains the true outcome with probability  $1 - \alpha$  on average across the population. That is, the coverage is right on average, but not necessarily for each individual. In practice one often desires *conditional coverage*; i.e., asking that the coverage guarantee holds not only on average but also for specific subpopulations or feature values.

Achieving exact conditional coverage is impossible in general without strong distributional assumptions (Vovk, 2012; Lei and Wasserman, 2014; Foygel Barber et al., 2021), and even approximate versions are notoriously difficult to deploy. Improving conditional coverage in CP typically requires carefully designed nonconformity scores. Common strategies rely on models that provide uncertainty estimates, such as quantile regression (Romano et al., 2019), predictive distributions (Izbicki et al., 2022; Braun et al., 2025), or local score adjustments (Guan, 2023; Messoudi et al., 2022; Thurin et al., 2025). Recent work proposes post hoc corrections that directly model conditional quantiles of the nonconformity score (Plassier et al., 2025). There is, however, a difficulty in assessing whether progress is being made in this literature, which is the lack of standard way to *evaluate* conditional coverage and thereby compare algorithms. The fundamental problem of the evaluation of conditional coverage is the focus of the current paper.

Evaluating conditional coverage is difficult. Group-based diagnostics, such as fairness-style coverage gaps (Ding et al., 2023), require large sample sizes per group and are highly sensitive to group definitions. Geometric scans such as worst-case slab coverage (WSC, Cauchois et al., 2021) offer a more adaptive view but suffer from severe sample complexity in high dimensions. Dependence-based diagnostics (Feldman et al., 2021) capture correlations between coverage and auxiliary variables but do not provide a standalone notion of conditional validity. In short, there is still no robust, general-purpose metric for assessing conditional coverage in practice.

**Contributions.** We address this gap by reframing conditional coverage evaluation as a supervised prediction task: given features  $X \in \mathcal{X}$ , predict whether the label  $Y \in \mathcal{Y}$  falls inside the predictive set  $C_\alpha(X)$ , where  $(X, Y) \sim \mathbb{P}_{X,Y}$ . Under perfect conditional coverage, for any proper loss  $\ell$  the Bayes-optimal predictor,  $h : \mathcal{X} \rightarrow \mathcal{Y}$ , which is defined as the minimizer of the risk  $\mathbb{E}[\ell(h(X), \mathbb{1}\{Y \in C_\alpha(X)\})]$  is the constant  $1 - \alpha$ . Consequently, any predictor that consistently outperforms this constant directly exposes a violation of conditional coverage. Building on this insight, we introduce the *excess risk of the target coverage* ( $\ell$ -ERT) metric to quantify deviations from conditional validity. Our metric provides an estimate of  $\mathbb{E}_X[D_\varphi(p(X)\|1-\alpha)]$  where  $D_\varphi$  is the Bregman divergence of a convex function  $\varphi$  (Bregman, 1967). For instance, we can reliably estimate the quantity  $\mathbb{E}[|1 - \alpha - \mathbb{P}(Y \in C_\alpha(X)|X)|]$ .

To contextualize our contribution, we establish a formal connection between existing group-based diagnostics and our metrics, showing that our formulation generalizes partition-based estimators to arbitrary predictor classes. This unified perspective unlocks the full potential of functional estimation, integrating both parametric and nonparametric approaches to assess conditional coverage, moving beyond heuristic group evaluations toward principled, model-based inference.

Through experiments, we demonstrate that our proposed conditional coverage metrics are empirically more robust, that is, less prone to misleading diagnostics, than existing alternatives. Finally, we benchmark several conformal prediction methods on real-world regression and classification tasks to compare their conditional coverage performance. To support reproducibility and to catalyze further progress in conformal prediction, we release `covmetrics`<sup>1</sup>, an open-source package for evaluating conditional coverage using our approach alongside established metrics.

Overall, this work casts *conditional coverage evaluation* as a key missing piece for practical conformal prediction, and offers new tools to fill the gap.

**Background on conformal prediction.** We begin with a brief overview of *split conformal prediction*, a widely used and computationally simple approach for constructing prediction sets with marginal validity guarantees (Papadopoulos et al., 2002; Lei et al., 2018; Angelopoulos and Bates, 2023). Suppose we observe data  $\mathcal{D} = \{(X_i, Y_i)\}_{i=1}^n$  sampled i.i.d. from a joint distribution  $\mathbb{P}_{X,Y}$ , where  $X_i \in \mathcal{X}$  are feature vectors and  $Y_i \in \mathcal{Y}$  are outcomes. For a new test pair,  $(X_{\text{test}}, Y_{\text{test}})$ , the aim is to form a predictive set  $C_\alpha(X_{\text{test}})$  such that

$$\mathbb{P}_{X,Y}(Y_{\text{test}} \in C_\alpha(X_{\text{test}})) = 1 - \alpha.$$

To achieve this, the dataset is randomly divided into two parts: a *training set*  $\mathcal{D}_1$  of size  $n_1$  and a *calibration set*  $\mathcal{D}_2$  of size  $n_2$ . A predictive model is fit on  $\mathcal{D}_1$ , while  $\mathcal{D}_2$  is reserved for calibrating the

---

<sup>1</sup><https://github.com/ElSacho/covmetrics>

prediction sets. Central to the procedure is a nonconformity score  $S(X, Y) \in \mathbb{R}$ , which quantifies how atypical a candidate response  $Y$  is relative to the model. Using these scores, the predictive sets are then computed as:

$$C_\alpha(X_{\text{test}}) = \{y : S(X_{\text{test}}, y) \leq \hat{q}_\alpha\}, \quad (1)$$

with

$$\hat{q}_\alpha := Q_{1-\alpha}\left(\frac{1}{n_2+1} \sum_{k=1}^{n_2} \delta_{S(X_k, Y_k)} + \frac{\delta_\infty}{n_2+1}\right), \quad (2)$$

where  $Q_{1-\alpha}(\mathbb{P})$  returns the  $1 - \alpha$  quantile of the distribution  $\mathbb{P}$ , and  $\delta_x$  is the Dirac measure centered at  $x$ . By the exchangeability of the data, this construction guarantees the marginal coverage property:

$$\mathbb{P}_{X,Y}(Y_{\text{test}} \in C_\alpha(X_{\text{test}}) \mid \mathcal{D}_1) \in \left[1 - \alpha, 1 - \alpha + \frac{1}{n_2+1}\right).$$

It is important to note that the guarantee in Eq. (1) is marginal, which means that coverage holds on average over the distribution of  $X_{\text{test}}$  and  $\mathcal{D}_2$ . A stronger requirement is *conditional coverage*, which demands

$$\mathbb{P}_{Y|X}(Y_{\text{test}} \in C_\alpha(X_{\text{test}}) \mid X_{\text{test}}) = 1 - \alpha, \quad (3)$$

for almost every  $X_{\text{test}}$ , but achieving (3) is impossible in general without additional assumptions. For a given conformal prediction strategy that achieves marginal coverage (1), it is essential to be able to measure how close from a conditional guarantee (3) we are.

**Notation.** We denote by  $\mathbb{1}_{x \in A}$  the indicator function, equal to 1 if  $x \in A$  and to 0 if  $x \notin A$ , for some set  $A$ . We denote by  $\text{sgn}(x)$  the function that returns the sign of  $x \in \mathbb{R}$ , where  $\text{sgn}(0) = 0$ . We write  $\Delta_d := \{p \in \mathbb{R}^d \mid p_i \geq 0 \forall i = 1, \dots, d, \sum_{i=1}^d p_i = 1\}$  to denote the probability simplex.

## 2 Related Work on Evaluating Conditional Coverage

In the following, we assume that a predictive model has already been trained to produce a predictive set rule  $C_\alpha(\cdot)$  for the output variable  $Y \in \mathcal{Y}$ , given a feature vector  $X \in \mathcal{X}$ . We are given a test dataset,  $\mathcal{D}_{\text{test}} = \{(X_i, Y_i)\}_{i=1}^m$ , to evaluate the conditional coverage of this predictive strategy. The test samples are assumed to be sampled i.i.d. from the distribution  $\mathbb{P}_{X,Y}$  and unseen during training.

**Group-based diagnostics.** A common way to study conditional coverage for a predictive set  $C_\alpha(\cdot)$  is to evaluate coverage over subpopulations or other partitions of the data. To make this concrete, fix a finite set of groups  $\mathcal{G}$  and a mapping  $g : \mathcal{X} \rightarrow \mathcal{G}$  that assigns each feature  $x \in \mathcal{X}$ , to a group  $g(x) \in \mathcal{G}$ . For a given group  $\mathbf{g} \in \mathcal{G}$  and a test sample with indices  $\mathcal{I}_{\mathbf{g}} = \{i : g(X_i) = \mathbf{g}\}$ , the empirical coverage in group  $\mathbf{g}$  is

$$C_{\mathbf{g}} = \frac{1}{|\mathcal{I}_{\mathbf{g}}|} \sum_{i \in \mathcal{I}_{\mathbf{g}}} \mathbb{1}\{Y_i \in C_\alpha(X_i)\}.$$

We review several strategies for defining these groups in Appendix A. Unless stated otherwise, in the following, the groups are obtained by clustering the feature space using the k-means algorithm. Below we review strategies that use such groupings to diagnose conditional miscoverage.

- **Coverage gap (CovGap).** This measure the average absolute deviation from the target coverage across groups,

$$\text{CovGap} = \frac{1}{|\mathcal{G}|} \sum_{\mathbf{g} \in \mathcal{G}} |C_{\mathbf{g}} - (1 - \alpha)|.$$

This metric is one of the most commonly used metrics in the literature (see, e.g, Ding et al. 2023; Kaur et al. 2025; Zhu et al. 2025; Filioux et al. 2024; Liu et al. 2025). A problem is that CovGap requires a large number of samples within each group to be consistent.

- **Weighted coverage gap (WCovGap).** To connect CovGap to our main metrics, we introduce its corrected version that assigns a weight to each group’s coverage gap:

$$\text{WCovGap} = \sum_{\mathbf{g} \in \mathcal{G}} \frac{|\mathcal{I}_{\mathbf{g}}|}{m} |C_{\mathbf{g}} - (1 - \alpha)|.$$

This formulation highlights that this metric can be interpreted as nonparametric estimators of the quantity,

$$\text{L}_1\text{-Miscoverage} := \mathbb{E}_X [ |\mathbb{P}_{Y|X}(Y \in C_{\alpha}(X) | X) - (1 - \alpha)| ].$$

Indeed,  $\text{CovGap}(X) := C_{g(X)}$  is an estimate of  $\mathbb{P}(Y \in C_{\alpha}(X) | g(X) = \mathbf{g})$ , and  $\frac{|\mathcal{I}_{\mathbf{g}}|}{m}$  an estimate of  $\mathbb{P}_X(g(X) = \mathbf{g})$ . Under standard regularity assumptions, specifically, if the partition  $\mathcal{G}$  becomes increasingly fine (i.e., the number of groups tends to infinity while their diameters shrink to zero) and if  $h^*(X) := \mathbb{E}[1_{Y \in C_{\alpha}(X)} | X]$  is Lipschitz-continuous, then the groupwise coverage  $C_{\mathbf{g}}$  converges to the true conditional coverage  $\mathbb{P}_{Y|X}(Y \in C_{\alpha}(X) | X)$  (see, e.g, [Györfi et al. 2005](#); [Bach 2024](#)). Consequently, the weighted CovGap (WCovGap) metric provides a nonparametric estimate of the stated quantity. If the groups are balanced in size, the metric CovGap admits the same type of probabilistic interpretation.

This observation is central to understanding the positioning of our work: previous strategies can be seen as partition-wise estimators of conditional coverage. In contrast, our approach leverages modern classifiers to obtain a more accurate estimation of conditional coverage.

**Worst-case slab diagnostic.** Rather than pre-specified groups, some diagnostics scan geometric slices of the feature space, if  $\mathcal{X} \subset \mathbb{R}^d$ . This strategy is commonly refer to as the worst-case slab coverage (WSC, [Cauchois et al., 2021](#)).

- **Worst-case slab coverage (WSC).** For a direction  $v \in \mathbb{R}^d$  and scalars  $a < b$ , define the slab

$$S_{v,a,b} := \{x \in \mathbb{R}^d : a \leq v^{\top} x \leq b\}.$$

For a mass threshold  $\delta \in (0, 1]$  let  $\mathcal{I}_{v,a,b} = \{i : X_i \in S_{v,a,b}\}$ . The empirical WSC in direction  $v$  is

$$\text{WSC}_n(C_{\alpha}(\cdot), v) := \inf_{a < b} \left\{ \frac{1}{|\mathcal{I}_{v,a,b}|} \sum_{i \in \mathcal{I}_{v,a,b}} \mathbb{1}\{Y_i \in C_{\alpha}(X_i)\} \middle| \frac{|\mathcal{I}_{v,a,b}|}{n} \geq \delta \right\}.$$

In practice, the induced metric requires a finite set of directions  $V$  and computes:

$$\text{WSC} = \inf_{v \in V} \text{WSC}_n(C_{\alpha}(\cdot), v).$$

This set is typically generated by sampling vectors at random from  $\mathbb{R}^d$ . When evaluating over a finite set of directions  $V$ , WSC uniformly approximates the population slab-coverage with high probability; the approximation error depends on the VC dimension ([Vapnik, 2000](#)) of the class of slabs induced by  $V$ . However, under conditional coverage, without sufficient test data WSC tends to provide a pessimist estimate of the conditional coverage violation by overfitting the test set by isolating miscovered points. Furthermore, this strategy does not adapt well to categorical data.

In Appendix A we review additional metrics. One of them is feature-stratified coverage (FSC), a group-based metric that reports the group with the worst coverage. This metric often appears in fairness-related work (see e.g, [Angelopoulos and Bates 2023](#); [Ding et al. 2023](#); [Jung et al. 2023](#)). Several grouping strategies besides categorical attributes or clustering have also been studied. For example, equal opportunity of coverage (EOC) ([Wang et al., 2023](#)) forms groups based on the output, and size-stratified coverage (SSC) ([Angelopoulos et al., 2021](#)) groups examples by the size of the prediction set. Another approach is to avoid explicit grouping and instead measure statistical dependence between the coverage indicator  $Z := \mathbf{1}\{Y \in C_{\alpha}(X)\}$  and the prediction-set size. [Feldman et al. \(2021\)](#) introduced

two such dependence measures based on Pearson’s correlation and the Hilbert–Schmidt independence criterion (HSIC).

Each diagnostic has strengths and limitations. Group-based metrics (CovGap, FSC, EOC, SSC) are intuitive and directly tied to fairness-style guarantees, but their statistical power depends strongly on the choice of groups and on having enough data per group. Geometric scans like WSC explore slices of  $\mathcal{X}$  without pre-specified semantic groups but suffer from the complexity of the feature space. Representation-based measures (Pearson, HSIC) provide complementary, model-driven checks for dependence between coverage and auxiliary signals, but low dependence does not prove full conditional coverage. A central challenge in catalyzing research progress in conformal prediction is the lack of reliable ways to assess conditional coverage empirically. Although many recent methods are designed to improve conditional coverage (Gibbs et al., 2025; Ding et al., 2023; Kaur et al., 2025; Plassier et al., 2025), existing guarantees are largely theoretical, and robust practical metrics remain elusive.

### 3 Evaluating Conditional Coverage

We would like the conditional coverage

$$p(x) := \mathbb{P}(Y \in C_\alpha(X) \mid X = x),$$

to be equal to  $1 - \alpha$   $\mathbb{P}_X$ -almost surely. Introducing the binary random variable  $Z = \mathbb{1}\{Y \in C_\alpha(X)\}$ , we can rewrite

$$p(x) = \mathbb{P}(Z = 1 \mid X = x).$$

Estimating  $\mathbb{P}(Z = 1 \mid X = x)$  is a binary classification problem, as we have access to a dataset of pairs  $(X_i, Z_i)$ . Some metrics such as CovGap or WSC implicitly learn classifiers based on histograms or slabs. However, these methods are rarely used for classification due to their poor practical performance. Explicitly reformulating conditional miscoverage estimation as a classification problem allows us to leverage strong and practically proven classifiers. Once a classifier  $h : \mathcal{X} \rightarrow [0, 1]$  is trained, we still need to use it to assess conditional miscoverage. The key idea is that under conditional coverage, given a proper score  $\ell$ , no classifier can achieve a lower risk than the constant predictor  $1 - \alpha$ . If we can learn a predictor that performs better, then conditional coverage does not hold. This leads to a metric with theoretical guarantees and a clear interpretation. In particular, our metric is a conservative estimate of  $\mathbb{E}[d(1 - \alpha, p(X))]$  for any  $d : [0, 1] \times [0, 1] \rightarrow \mathbb{R}$  such that for all  $p \in [0, 1]$ ,  $d(p, \cdot)$  is convex and minimized at  $p$ .

#### 3.1 Excess risk of the target coverage (ERT)

For a given classifier  $h$  and loss function  $\ell$ , the associated risk is defined as

$$\mathcal{R}_\ell(h) := \mathbb{E}_{X, Z}[\ell(h(X), Z)].$$

The Bayes predictor in this task is (see, e.g., Devroye et al. 2013):

$$h^*(x) \in \operatorname{argmin}_{q \in [0, 1]} \mathbb{E}[\ell(q, Z) \mid X = x].$$

If the loss  $\ell$  is a proper loss (see, e.g, Gneiting and Raftery 2007; Bröcker 2009), then it is optimal to predict the true probability  $h^*(X) = \mathbb{E}[Z \mid X] = p(X)$   $\mathbb{P}_X$ -almost surely. If conditional coverage holds, we get  $h^*(X) = \mathbb{E}[Z \mid X] = 1 - \alpha$   $\mathbb{P}_X$ -almost surely, so no classifier can achieve lower risk than the constant  $1 - \alpha$  prediction. Proper losses include the Brier score,  $\ell(p, y) = (p - y)^2$ , and the log-loss score,  $\ell(p, y) = -y \log p - (1 - y) \log(1 - p)$ , where  $p \in [0, 1]$  denotes the predicted probability of the event occurring and  $y \in \{0, 1\}$  denotes the observed outcome.

This motivates the *excess risk of the target coverage* ( $\ell$ -ERT). For a general proper loss  $\ell$ , we define

$$\ell\text{-ERT} := \mathcal{R}_\ell(1 - \alpha) - \mathcal{R}_\ell(p).$$

Larger values of  $\ell$ -ERT correspond to greater violations of conditional coverage.

**Interpretation and examples.** ERT has a probabilistic interpretation. Indeed,

$$\begin{aligned}
\ell\text{-ERT} &= \mathcal{R}_\ell(1 - \alpha) - \mathcal{R}_\ell(p) \\
&= \mathbb{E}_{X, Z}[\ell(1 - \alpha, Z) - \ell(p(X), Z)] \\
&= \mathbb{E}_X[\mathbb{E}_Z[\ell(1 - \alpha, Z) - \ell(p(X), Z)|X]] \\
&= \mathbb{E}_X[d_\ell(1 - \alpha, p(X))],
\end{aligned}$$

where  $d_\ell(p, q) := \mathbb{E}_{y \sim q}[\ell(p, y) - \ell(q, y)]$  is the divergence associated with the proper score  $\ell$  (see, e.g, Bröcker 2009). We summarize the  $\ell$ -ERT scores for different proper scores in Table 1.

Name	Proper score $\ell(p, y)$	$\ell$ -ERT formula
$L_1$ -ERT	$\text{sgn}(p - (1 - \alpha))(1 - \alpha - y)$	$\mathbb{E}_X[ 1 - \alpha - p(X) ]$
$L_2$ -ERT	Brier score: $(y - p)^2$	$\mathbb{E}_X[(1 - \alpha - p(X))^2]$
KL-ERT	Log-loss: $-\log p_y$	$\mathbb{E}_X[D_{\text{KL}}(p(X) \  1 - \alpha)]$

Table 1: Examples of proper scoring rules and their associated ERT scores.

We will show in Section 3.2 that a general class of convex distances can be estimated via ERTs.

**Estimation from finite samples.** Since  $p(X)$  is unknown in practice, we define the functional

$$\ell\text{-ERT}(h) := \mathcal{R}_\ell(1 - \alpha) - \mathcal{R}_\ell(h), \quad (4)$$

This metric quantifies how much better a predictor  $h$  performs relative to the constant baseline  $1 - \alpha$ . While we cannot guarantee that the learned predictor  $h$  coincides with the Bayes-optimal predictor  $h^*$ , our procedure always provides a lower bound on the true  $\ell$ -ERT, that is for all measurable classifiers,

$$\ell\text{-ERT}(h) \leq \ell\text{-ERT}.$$

Therefore, it suffices to find an  $h$  that performs better than the constant  $1 - \alpha$  to conclude that conditional coverage is not achieved, and use  $\ell\text{-ERT}(h)$  to lower-bound  $\mathbb{E}[d_\ell(1 - \alpha, p(X))]$ .

To estimate  $\ell\text{-ERT}(h)$ , we can evaluate their empirical risks that is

$$\widehat{\ell\text{-ERT}}(h) = \frac{1}{m} \sum_{i=1}^m [\ell(1 - \alpha, Z_i) - \ell(h(X_i), Z_i)].$$

To avoid overfitting and misleading diagnostics, we cannot train  $h$  on the values  $X_i$  that it is evaluated on. In general, cross-validation can be used for this purpose, where multiple classifiers are trained on different subsets of the data, such that each data point can be evaluated using a classifier that was not trained on it. For random forest, we can use out-of-bag predictions. The resulting algorithm for evaluating conditional coverage using  $k$ -fold cross-validation is summarized in Algorithm 1.

Figure 1 illustrates the usefulness of our metric by showing prediction sets produced under different conformal strategies together with their estimated conditional coverage. The function  $h$  is estimated with a neural network that has two hidden layers of width 64. In the first strategy, which applies a non conditional conformal method, the prediction sets fail to reflect local variations in conditional miscoverage. This leads to large estimated ERT values, with  $L_1\text{-ERT}(h) \approx 0.0757$  and  $L_2\text{-ERT}(h) \approx 0.0073$ . In the second strategy, where prediction sets are closer to satisfying conditional coverage, the estimator  $h$  identifies coverage levels near  $1 - \alpha$ . The resulting ERT values are closer to zero, with  $L_1\text{-ERT}(h) \approx 0.0148$  and  $L_2\text{-ERT}(h) \approx -0.00002$ , which signals improved conditional behavior.

We also compare our functional estimator  $h$  to the partition based nonparametric estimator  $\text{CovGap}(X)$ . In one dimension, the feature space is simple to cluster and the partition-based approach can approximate  $p$  well. This advantage does not persist as the feature dimension grows, a point that will be demonstrated in Section 4.

---

**Algorithm 1** Compute  $\widehat{\ell\text{-ERT}}$ .

---

**Require:** Data  $\{(X_i, Z_i)\}_{i=1}^m$ , number of folds  $k \geq 2$ , proper score  $\ell$ , level  $\alpha$ , classification method.

1: **Partition the data:** Randomly divide  $\{1, \dots, m\}$  into  $k$  approx. equal-sized folds  $\{\mathcal{I}_1, \dots, \mathcal{I}_k\}$ .

2: **for**  $j = 1$  to  $k$  **do**

3:   **Define folds:**  $\mathcal{I}_{\text{val}}^{(j)} = \mathcal{I}_j$ ,  $\mathcal{I}_{\text{tr}}^{(j)} = \{1, \dots, m\} \setminus \mathcal{I}_j$ .

4:   **Train classifier:** Fit a classifier  $h^{(j)}$  on  $\{(X_i, Z_i) \mid i \in \mathcal{I}_{\text{tr}}^{(j)}\}$  using the specified method.

5:   **Evaluate on validation fold:** For each  $i \in \mathcal{I}_{\text{val}}^{(j)}$ , compute

$$\widehat{\ell\text{-ERT}}^{(j)} = \frac{1}{|\mathcal{I}_{\text{val}}^{(j)}|} \sum_{i \in \mathcal{I}_{\text{val}}^{(j)}} \ell(1 - \alpha, Z_i) - \ell(h^{(j)}(X_i), Z_i).$$

6: **end for**

7: **Aggregate across folds:**  $\widehat{\ell\text{-ERT}} = \frac{1}{k} \sum_{j=1}^k \widehat{\ell\text{-ERT}}^{(j)}$ .

---

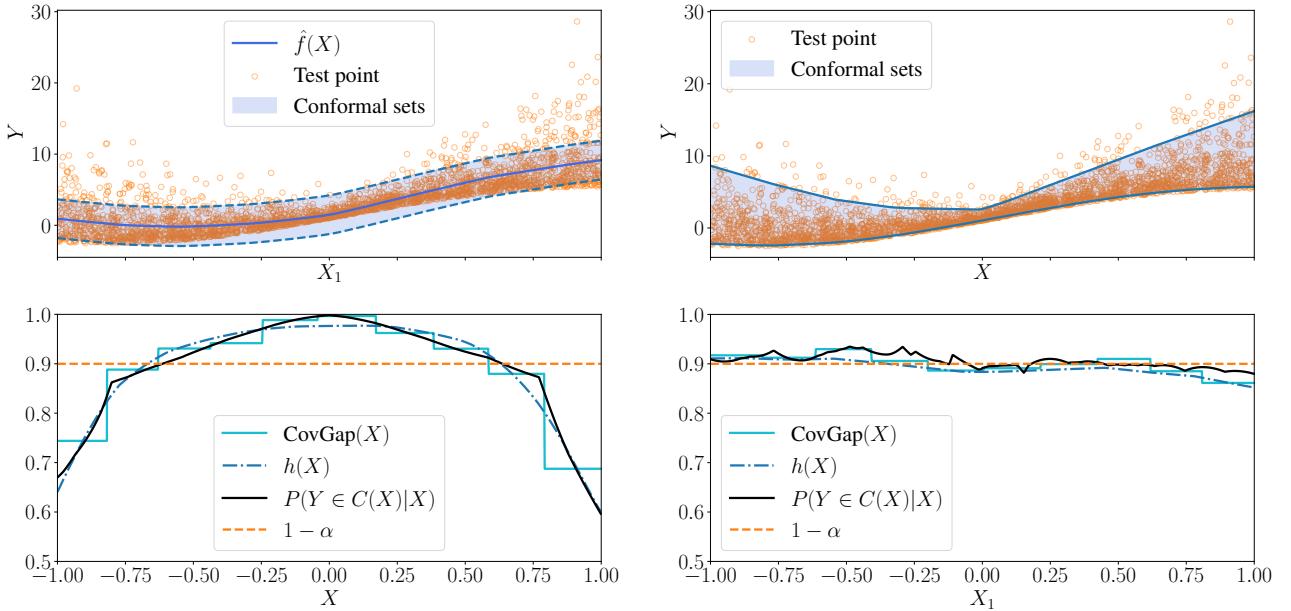


Figure 1: Illustration of conditional coverage estimation. The top panel shows data generated from  $Y \sim \mathcal{N}(f(X), \sigma(X))$ ,  $X \sim \mathcal{U}([-1, 1])$  with  $f(x) = 3 \sin(x) + e^x$  and  $\sigma(x) = 1/2 + |x| + x^2$ , and their predictive sets. The bottom panel shows the conditional coverage estimation  $h$  used to estimate the  $L_2$ -ERT( $h$ ), conditional coverage estimation induced by a partition-wise estimator, true conditional coverage, and desired  $1 - \alpha$  conditional coverage. **Left:** Conformal sets from the score  $S(X, Y) = |Y - \hat{f}(X)|$ . **Right:** Conformal sets by fitting quantiles  $\alpha/2$  and  $1 - \alpha/2$  of  $\mathbb{P}_{Y|X}$  following the procedure in [Romano et al. \(2019\)](#).

### 3.2 Estimating general distances

Previously, Table 1 illustrated that specific choices of the proper loss  $\ell$  can recover common distance functions. However, we can go much beyond that and estimate any convex distance function  $f(q) = d(1 - \alpha, q)$  using an ERT, as long as the proper score  $\ell$  is allowed to depend on  $f$  and therefore the coverage  $1 - \alpha$  itself. The following proposition formalizes this statement.

**Proposition 3.1** (Representing convex losses as ERTs). *Let  $f : [0, 1] \rightarrow \mathbb{R}_{\geq 0}$  be convex with  $f(1 - \alpha) = 0$ . Let  $f'$  be a subderivative of  $f$  satisfying  $f'(1 - \alpha) = 0$ . Then, the function*

$$\ell(p, y) := \ell_{f, f'}(p, y) := -f(p) - (y - p)f'(p) \quad (p \in [0, 1], y \in \{0, 1\})$$

is a proper score satisfying

$$\ell\text{-ERT} = \mathbb{E}_X[f(p(X))] .$$

*Proof.* First,  $\ell$  is a proper score because for all  $p, q \in [0, 1]$ , convexity of  $f$  yields

$$\mathbb{E}_{y \sim q}[\ell(p, y)] = -f(p) - (q - p)f'(p) \geq -f(q) = \mathbb{E}_{y \sim q}[\ell(q, y)] .$$

Since we assumed  $f'(1 - \alpha) = 0$ , we have  $\ell(1 - \alpha, Z) = -f(1 - \alpha) = 0$ . Hence,

$$\mathbb{E}_{Z \sim p}[\ell(1 - \alpha, Z) - \ell(p, Z)] = \mathbb{E}_{Z \sim p}[-\ell(p, Z)] = f(p) + (p - p)f'(p) = f(p) .$$

Taking the expectation over  $X$  yields the claim.  $\square$

A related formulation in Section B shows that ERT can estimate  $d$  if it is a Bregman divergence of convex functions.

### 3.3 Separating over-coverage and under-coverage

Theorem 3.1 implies that we can estimate asymmetric distance measures to gain more insights on the nature of miscoverage. In particular, one can decompose the convex function  $f$  from above as  $f = f_+ + f_-$  with an over-coverage part  $f_+(p) := f(\max\{p, 1 - \alpha\})$  that only penalizes the case  $p > 1 - \alpha$  and an under-coverage part  $f_-(p) = f(\min\{p, 1 - \alpha\})$  that only penalizes  $p < 1 - \alpha$ . Correspondingly, one can decompose the proper loss  $\ell$  and the ERT as

$$\begin{aligned} \ell(p, y) &= \ell_+(p, y) + \ell_-(p, y) - \ell(1 - \alpha, y) \\ \ell\text{-ERT} &= \ell_+\text{-ERT} + \ell_-\text{-ERT} \\ \ell_+(p, y) &:= \ell(\max\{p, 1 - \alpha\}, y) \\ \ell_-(p, y) &:= \ell(\min\{p, 1 - \alpha\}, y) . \end{aligned}$$

Together,  $\ell_+\text{-ERT}$  and  $\ell_-\text{-ERT}$  provide a decomposition of conditional coverage error into two complementary components. The first identifies unnecessary conservatism, while the second highlights locations where the procedure is too aggressive and exhibits under-coverage. This split view delivers more informative diagnostics and supports targeted improvements in the design of conformal prediction methods.

### 3.4 Extensions

**A proxy for conditional coverage.** The learned predictor  $h$  can also be used as a proxy for conditional coverage. For a given test point  $X_{\text{test}}$ , its conditional coverage can be estimated as

$$h(X_{\text{test}}) \approx \mathbb{P}(Y_{\text{test}} \in C_\alpha(X_{\text{test}}) \mid X_{\text{test}}).$$

This approximation provides a corrective proxy for conditional coverage: rather than modifying the predictive set  $C_\alpha(X_{\text{test}})$ , we adjust its predicted coverage level from  $1 - \alpha$  to  $h(X_{\text{test}})$ .

**Evaluating conditional coverage rules.** We further extend our metric to settings where the target conditional coverage is not fixed at  $1 - \alpha$ , but instead varies according to a specified decision rule. This extension, detailed in Appendix D, enables testing whether a given strategy satisfies conditional coverage with respect to adaptive or context-dependent coverage levels.

## 4 Experiments

Our package `covmetrics`<sup>2</sup> and the code for all our experiments<sup>3</sup> is accessible and reproducible from GitHub.

<sup>2</sup><https://github.com/ElSacho/covmetrics>

<sup>3</sup>[https://github.com/ElSacho/Conditional\\_Coverage\\_Estimation](https://github.com/ElSacho/Conditional_Coverage_Estimation)

## 4.1 Comparing different classifiers

The quality of the ERT estimation hinges on the choice of a good classifier, which depends on the data type of  $X$ . We will restrict our experiments to the case where  $X$  is a fixed-dimensional vector of numerical and/or categorical features, also known as *tabular data*. For other modalities like images or text, tabular classifiers could be used on top of embeddings of  $X$  to keep the training fast. As we want our metric to be reasonably fast to compute, we are particularly interested in finding fast classifiers. For this reason, we do not tune the hyperparameters of the classifiers. Based on recent benchmarks (Erickson et al., 2025; Holzmüller et al., 2024), we choose a subselection of classifiers that are promising in terms of their speed-accuracy trade-off. We provide more details on those classifiers in Appendix F. We note that our results show performances of specific configurations, but other trade-offs can also be achieved.

We begin by examining how various classifiers perform when estimating the conditional coverage quantity. This is achieved by comparing how well different tabular classifiers estimate the conditional miscoverage. We review those classifiers in Appendix F. To pursue this, we select the four largest regression datasets in TabArena (Erickson et al., 2025). Each dataset is divided into three parts; a training set (with 40% of the data) used to learn a predictor  $f$  that minimizes the empirical mean squared error. A calibration set (with 10% of the data) used with the nonconformity score  $S(X, Y) = |Y - f(X)|$  to construct the set rule  $C_\alpha(\cdot)$  with  $1 - \alpha = 0.9$  such that when the residual distribution is heteroskedastic, these sets are not expected to be conditional. A test set (with 50% of the data) used to evaluate the  $L_1$ -ERT,  $L_2$ -ERT and KL-ERT metrics, performing 5-fold cross-validation.

We compare the estimated metric values as a function of the number of test samples and average all results over ten runs. Since our estimator gives a lower bound on the true  $\ell$ -ERT value, a larger estimate indicates a stronger classifier. In Table 2 we report the average percentage improvement over the best strategy, averaged across all test sample sizes and all datasets, as well as the average time required to estimate our metrics per 1,000 samples. Because averaging can hide important effects, we also report results for each dataset as a function of the number of test samples in Figures 2 and 3.

Classifier	Avg. % of max ERT			Avg. time per 1K samples [s]	Device
	$L_1$ -ERT	$L_2$ -ERT	KL-ERT		
TabICLv1.1	<b>71.9</b> <sub>1.9</sub>	55.4 <sub>2.7</sub>	<b>60.2</b> <sub>1.8</sub>	5.1 <sub>0.7</sub>	GPU
RealTabPFN-2.5	71.6 <sub>1.7</sub>	<b>55.7</b> <sub>2.5</sub>	59.7 <sub>1.9</sub>	5.2 <sub>0.7</sub>	GPU
CatBoost	68.7 <sub>2.8</sub>	50.8 <sub>2.7</sub>	55.2 <sub>2.1</sub>	18.7 <sub>4.5</sub>	CPU
LightGBM (medium)	68.4 <sub>2.2</sub>	49.9 <sub>2.5</sub>	53.6 <sub>1.7</sub>	2.6 <sub>0.3</sub>	CPU
ExtraTrees	65.9 <sub>2.4</sub>	30.5 <sub>3.2</sub>	0.0 <sub>0.0</sub>	2.1 <sub>0.4</sub>	CPU
RandomForest	65.9 <sub>2.8</sub>	35.7 <sub>2.5</sub>	0.0 <sub>0.0</sub>	2.7 <sub>0.5</sub>	CPU
PartitionWise	38.3 <sub>1.9</sub>	14.1 <sub>1.1</sub>	1.7 <sub>0.9</sub>	<b>0.2</b> <sub>0.0</sub>	CPU

Table 2: **ERT recovered by different methods**, relative to the highest value among all methods and number of samples, averaged over all number of test samples and datasets. Experiments are repeated 10 times, and the index number is the standard deviation across those 10 experiments.

**Results.** As shown in Table 2, the tabular foundation models TabICLv1.1 (Qu et al., 2025) and RealTabPFN-2.5 (Grinsztajn et al., 2025) show excellent performance across different metrics. However, they can be very slow without a GPU, and they are generally only usable up to a certain size of datasets (around 100K samples). Gradient-boosted decision trees like CatBoost (Prokhorenkova et al., 2018) and LightGBM (Ke et al., 2017) are closely behind, but they do facilitate fast training on CPUs and are scalable to large datasets. In particular, LightGBM with the relatively cheap configuration from Holzmüller et al. (2024) excels through its training speed while still recovering large ERT values. Therefore, we suggest LightGBM as the default classifier to use with ERT. Random Forest (Breiman, 2001) and ExtraTrees (Geurts et al., 2006) are also fast but exhibit worse results, particularly for the KL-ERT, which heavily penalizes overconfidence, and the  $L_2$ -ERT. PartitionWise, the strategy used

for CovGap, is fastest but performs much worse than the other classifiers. Figures 2 and 3 show that on two out of four datasets, PartitionWise detects almost no miscoverage. Overall, our results suggest that results of tabular classification benchmarks transfer approximately to the task of ERT estimation.

**Differences between metrics.** Table 2 also shows that the  $L_1$ -ERT is considerably easier to estimate than the  $L_2$ -ERT and KL-ERT. Indeed, for the  $L_1$ -ERT, the classifiers only need to predict on the right side of  $1 - \alpha$  to be optimal, whereas for the others they need to predict the exact probability. Hence, we recommend using the  $L_1$ -ERT as the default metric.

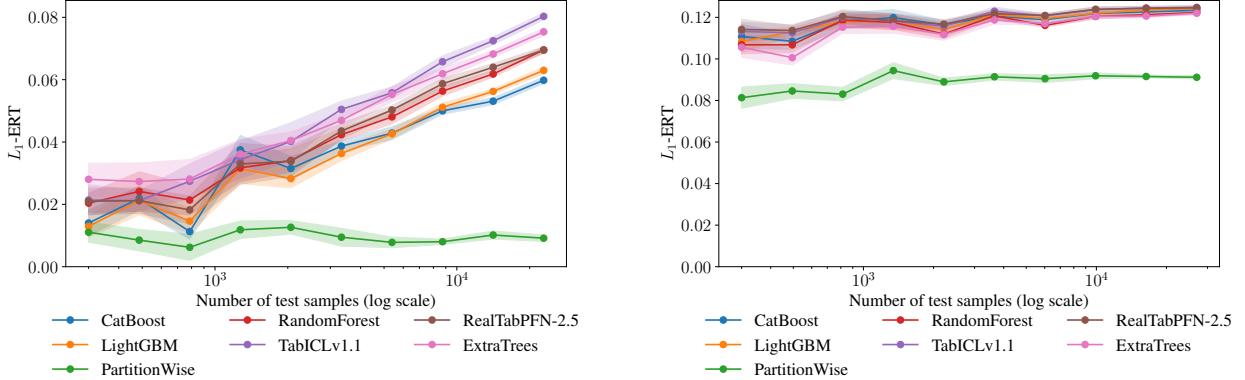


Figure 2: Illustration of the estimation of  $L_1$ -ERT for different classifiers as a number of sampled data available. **Left:** physicochemical\_protein dataset. **Right:** Diamonds dataset.

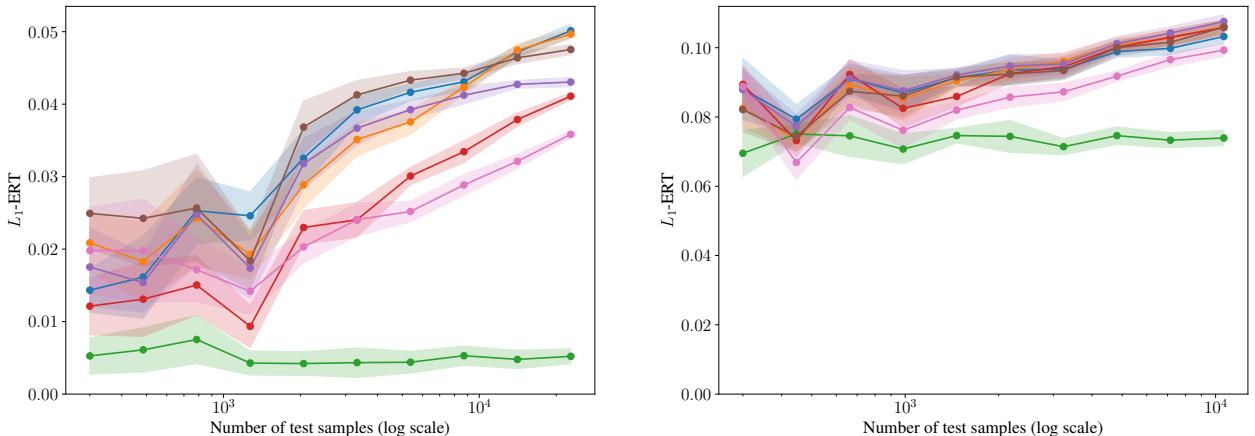


Figure 3: Illustration of the estimation of  $L_1$ -ERT for different classifiers as a number of sampled data available. Legend shared with Figure 2. **Left:** Food\_Delivery\_Time dataset. **Right:** Superconductivity dataset.

## 4.2 Comparison with existing metrics

We illustrate that existing methods can fail to accurately assess conditional coverage in scenarios where our approach succeeds. To this end, we generate a synthetic dataset following

$$Y \sim \mathcal{N}(0, \sigma(X^1)), \quad \text{with} \quad X \sim \mathcal{U}([-1, 1]^8),$$

where  $X^1$  is the first component of the vector  $X$ .

Prediction sets are constructed using two different strategies:

- **Standard CP:** Using the nonconformity score  $S(X, Y) = |Y|$  within the standard conformal prediction framework using 3,000 i.i.d. samples.

- **Oracle sets:** Using the ground-truth oracle that provides the true conditional quantiles  $\alpha/2$  and  $1 - \alpha/2$  of the underlying distribution.

The first strategy produces marginally valid but conditionally invalid prediction sets, while the second produces conditional sets by construction.

The first experiment evaluates how many test points are needed to obtain reliable estimates for commonly used metrics (CovGap, WSC) compared to our proposed metrics. We measure each metric as a function of the number of test points on a log scale, computing  $L_1$ -ERT and  $L_2$ -ERT respectively estimating  $\mathbb{E}_X[|1 - \alpha - p(X)|]$  and  $\mathbb{E}_X[(1 - \alpha - p(X))^2]$  using 5-fold cross-validation, and show the results in Figure 4.

**Results.** The results are striking: group-based metrics are extremely unaligned with their theoretical values and require large sample sizes to converge. Even with 5,000 points, they provide nearly identical diagnostics across these two very different scenarios, and WSC exhibits similar instability. By contrast, our metrics adapt rapidly. In particular,  $L_1$ -ERT stabilizes very quickly, providing reliable estimates of conditional coverage deviation in the naive scenario.  $L_2$ -ERT needs more samples to converge to its true value but already diagnoses conditional coverage failure with few samples. This is not surprising as, as explained earlier, the  $L_1$  version only requires that the sign of  $h(X) - (1 - \alpha)$  matches the sign of  $p(X) - (1 - \alpha)$ , whereas the  $L_2$  version instead depends on the closeness of  $h(X)$  to the true conditional probability itself, which typically requires more data.

In the scenario with perfect conditional coverage, all of our proposed metrics converge rapidly to values indicating no failure, while WSC continues to struggle even with 50,000 samples. These results demonstrate that our methods not only provide more accurate diagnostics but also require far fewer samples to detect conditional coverage deviations reliably.

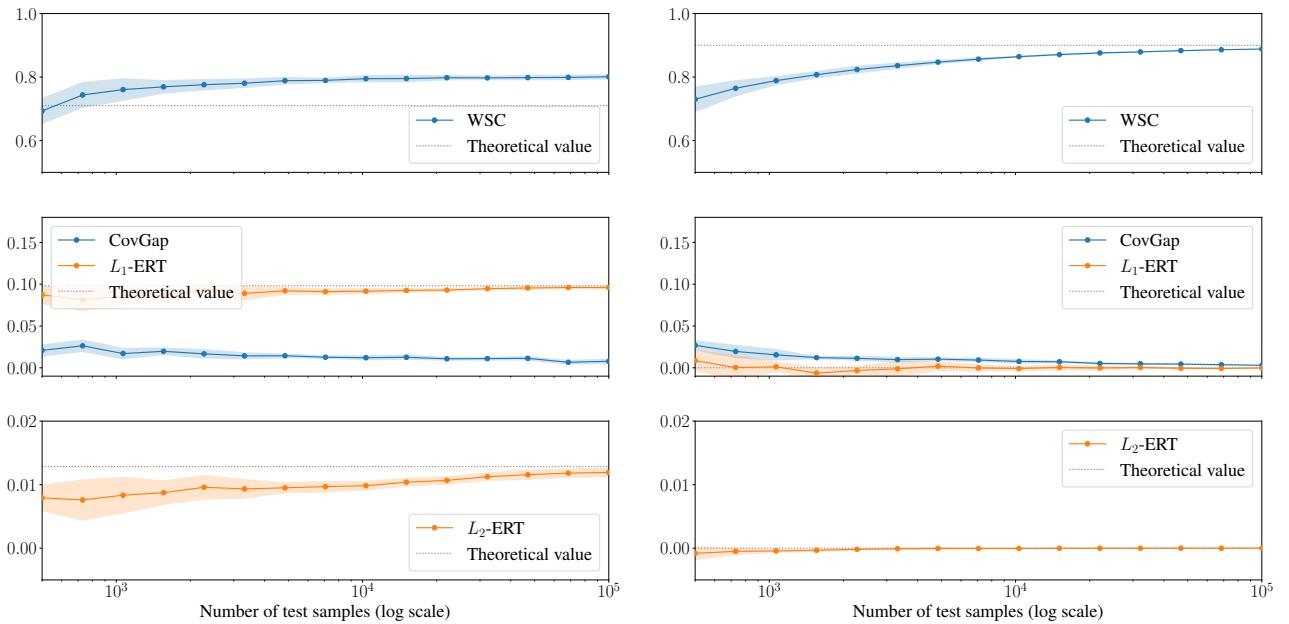


Figure 4: Estimated metrics as a number of test samples points. Top figure: WSC. Middle figure: CovGap &  $L_1$ -ERT. Bottom axis:  $L_2$ -ERT **Left:** Standard CP not conditional. **Right:** Oracle sets conditional. Theoretical values are estimated using the true value  $\mathbb{P}(Y \in C_\alpha(X) | X)$  for 300,000 samples of  $\mathbb{P}_X$ .

Figure 5 visualizes the data distribution and the induced prediction sets for both strategies with test dataset of size 1,500. Precise metrics values are reported in Table 3. Since only the first feature is informative, we plot  $(X^1, Y)$  while ignoring the remaining features, and we also show  $(X^1, h(X))$  to illustrate our estimator's learned conditional coverage, as well as  $(X^1, \text{CovGap}(X))$  to illustrate the difference between a partition-wise estimator and our estimator. As expected, our approach clearly

identifies regions of under- and over-coverage in the first scenario, and accurately recovers a near-constant predictor equal to  $1 - \alpha$  in the oracle setting.

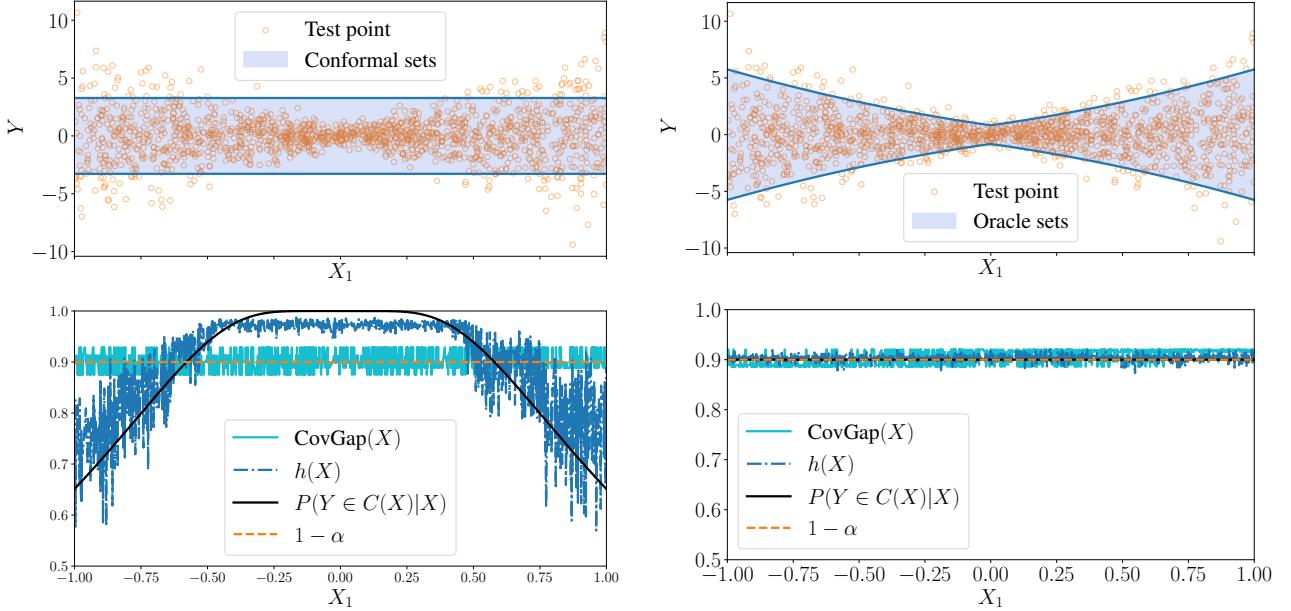


Figure 5: Illustration of conditional coverage estimation. The top panel shows data, generated from  $Y \sim \mathcal{N}(0, \sigma(X^1))$  with  $\sigma(x) = 0.5 + |x| + x^2$ , where  $X \sim \mathcal{U}([-1, 1]^8)$ , and  $X^1$  is the first component of such a vector  $X$ . The bottom panel shows the conditional coverage estimation  $h$ , conditional coverage estimation induced by a partition-wise estimator, true conditional coverage, and desired  $1 - \alpha$  conditional coverage. **Left:** Conformal sets from the score  $S(X, Y) = |Y|$  using 3,000 samples. **Right:** Oracle sets that achieve conditional coverage.

We evaluate the metrics on a test set of size 1,500, as reported in Table 3. In the first experiment, where the predictive sets are not conditional, certain metrics fail to detect deviations from conditional coverage. This is the case for SSC, HSIC, and Pearson correlation as they rely solely on prediction set sizes, which are uniform across samples. Similarly, metrics based on feature-space clustering (FSC, CovGap) also fail to detect the coverage violation, due to the difficulty of clustering the feature space. The only baseline that identifies a conditional coverage failure in this case is the WSC metric.

In contrast, under the second (oracle) strategy, all prediction sets satisfy conditional coverage by construction. The WSC metric still reports a conditional coverage failure, with values comparable to the first example. This occurs because, in high-dimensional feature spaces, WSC can overemphasize local fluctuations and misidentify regions with apparent over- or under-coverage. Similarly, EOC detects a false conditional coverage violation by grouping extreme outputs values together.

Our proposed  $L_1$ -ERT metric and  $L_2$ -ERT metrics, however, correctly distinguish between the two settings: they detect the conditional coverage failure in the first example and reports no such failure in the oracle case.

### 4.3 Real datasets

#### 4.3.1 Regression

We next compare conditional coverage metrics across the most widely used conformal prediction strategies. In particular, we build upon the benchmarking framework of Dheur et al. (2025) to evaluate multivariate regression conformal prediction methods. For each strategy, we first train the underlying predictive model and then conformalize its outputs using the standard split conformal prediction procedure. Detailed descriptions of all strategies are provided in Appendix E.

Our evaluation is conducted on several datasets commonly used in regression studies. Dataset specifications and algorithmic details are reported in Appendix G.1.

All experiments are repeated ten times, and we report averaged estimates across runs. We distinguish between univariate regression ( $Y \in \mathbb{R}$ ) and multivariate regression ( $Y \in \mathbb{R}^k$ , for  $k \geq 2$ ) to

	Not conditional	Conditional
FSC	0.868 <sub>0.014</sub>	<b>X</b> 0.881 <sub>0.012</sub> ✓
CovGap	0.016 <sub>0.004</sub>	<b>X</b> 0.014 <sub>0.004</sub> ✓
WCovGap	0.016 <sub>0.004</sub>	<b>X</b> 0.014 <sub>0.004</sub> ✓
WSC	0.740 <sub>0.019</sub>	✓ 0.790 <sub>0.014</sub> <b>X</b>
EOC	0.341 <sub>0.009</sub>	✓ 0.186 <sub>0.018</sub> <b>X</b>
SSC	0.004 <sub>0.003</sub>	<b>X</b> 0.013 <sub>0.003</sub> ✓
HSIC	0.000 <sub>0.000</sub>	<b>X</b> 0.000 <sub>0.000</sub> ✓
Pearson	0.000 <sub>0.000</sub>	<b>X</b> 0.019 <sub>0.016</sub> ✓
<i>L</i> <sub>1</sub> -ERT (Ours)	0.091 <sub>0.007</sub>	✓ −0.005 <sub>0.009</sub> ✓
<i>L</i> <sub>2</sub> -ERT (Ours)	0.009 <sub>0.001</sub>	✓ −0.000 <sub>0.000</sub> ✓

Table 3: Conditional metrics for both synthetic samples and whether they accurately diagnose conditional coverage. ✓: Accurate diagnostic. **X**: Failure.

highlight how conditional coverage behaves under increasing output dimensionality. The univariate results are deferred to Appendix H.

**Multivariate results.** We present aggregated results across six datasets, with dataset specific information provided in Appendix H. Our first analysis compares the metrics  $L_1$ -ERT and WCovGap, which both aim to estimate  $\mathbb{E}_X[|1 - \alpha - p(X)|]$ , as shown in Figure 6. Each metric is averaged over all datasets. Although both target the same quantity, our metric consistently yields a larger estimate of conditional miscoverage. Because it serves as a lower bound on the true expectation, this indicates that our approach offers a more precise view of conditional coverage than the partition based estimator used in prior methods. Figure 7 then compares the estimation of WSC with our estimator for  $L_2$ -ERT.

These results, however, must be interpreted with care. As shown in Figure 8, improvements in conditional coverage often come hand-in-hand with larger prediction sets. For instance, the C-PCP (Dheur et al., 2025) method achieves the best conditional coverage across all metrics, but also produces one of the largest prediction intervals. Conversely, methods such as MVCS (Braun et al., 2025) yield smaller predictive sets at the cost of poorer conditional coverage. This observation underscores a fundamental trade-off: strategies that aggressively minimize prediction volume while maintaining marginal coverage often sacrifice conditional coverage. Understanding and managing this trade-off is crucial for tailoring conformal prediction methods to specific applications.

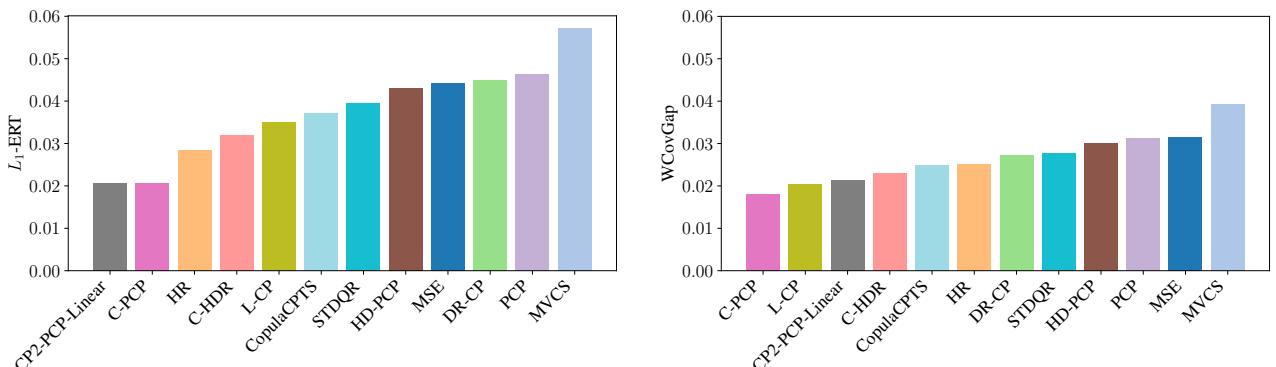


Figure 6: Metric values averaged across all datasets for all methods in multivariate regression. **Left:**  $L_1$ -ERT (lower is better). **Right:** WCovGap (lower is better)

#### 4.3.2 Classification

In classification, most of the strategies are commonly tailored to specific problems such as long-tailed classification, so we choose to only compare the two most used conformal prediction strategies for

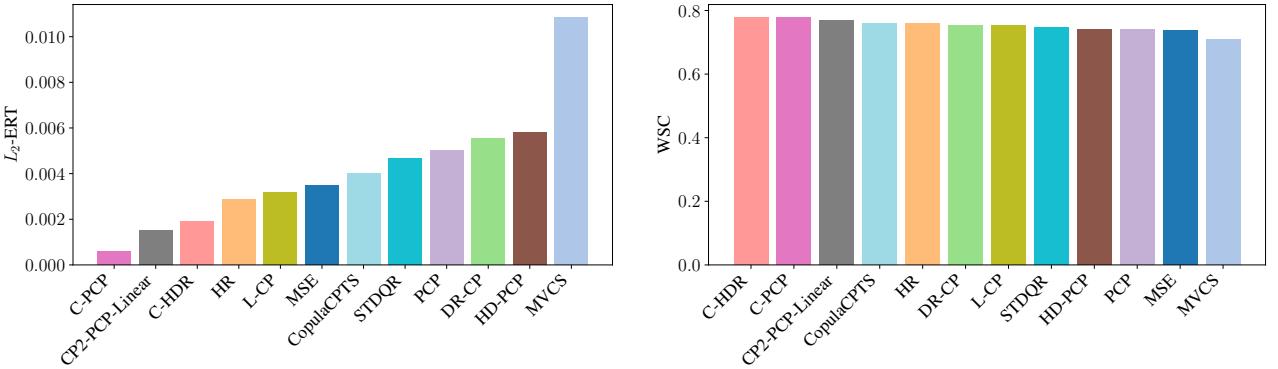


Figure 7: Metric values averaged across all datasets for all methods in multivariate regression. **Left:**  $L_2\text{-ERT}$  (lower is better). **Right:** WSC (closer to 0.9 is better).

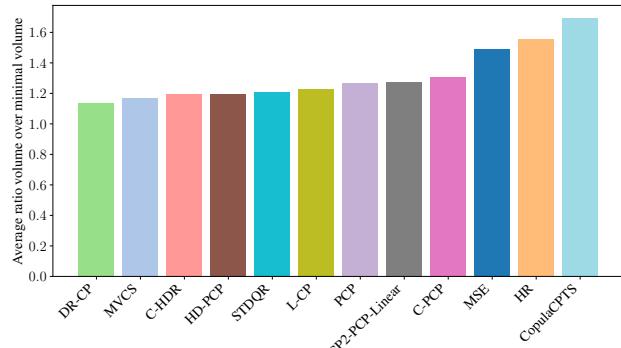


Figure 8: Normalized set sizes averaged all datasets in multivariate regression, where the normalization is done by dividing each volume by the smallest volume across all methods (smaller is better).

classification, given a predictive model that returns probabilities estimates  $\hat{f}(X) \in \Delta_d$ . The first one is the negative likelihood prediction (Sadinle et al., 2019) and uses the score  $S(X, Y) = -p(X)_Y$ . The second one (Romano et al., 2020; Angelopoulos et al., 2021) uses the cumulative likelihood scores. We first define the permutation  $\pi(x)$  of  $\{1, \dots, K\}$  that sorts the probabilities in decreasing order, i.e.,

$$\hat{f}_{\pi_1(x)}(x) \geq \hat{f}_{\pi_2(x)}(x) \geq \dots \geq \hat{f}_{\pi_K(x)}(x).$$

Then, the score function is defined as :

$$S(X, Y) = \sum_{j=1}^{\pi_k(X)} \hat{f}_{\pi_j(X)}(X), \quad \text{where } Y = \pi_k(X).$$

For MNIST, FashionMNIST, and CIFAR, we trained a CNN composed of two convolutional layers followed by max pooling, then two fully connected layers with dropout and ReLU activations. The model was trained with cross entropy loss to learn  $\hat{f}$ . We used early stopping when the accuracy fell below  $1 - \alpha$ , since otherwise both conformal strategies tend to produce many empty sets, which would make the results uninformative.

For the CIFAR100 experiment, we trained a ResNet model with cross-entropy loss to learn  $\hat{f}$ . To learn the classifier for ERT, we re-used this pretrained model, but replaced its final layer with a new one. This avoided the cost of learning a large feature space from scratch.

We report the ERT values in Table 4. For the classification problem, both strategies remain far from conditional. In general, we believe that calibrating the predictors leads to better conditional coverage. Interestingly for CIFAR100, the  $L_1$ -ERT and the KL-ERT lead to two different conclusions: the former suggests that the likelihood strategy is more conditional than the cumulative one, while the latter suggests the opposite. We attribute this discrepancy to the larger number of empty predictive sets produced by the likelihood strategy, for which the conditional coverage equals zero, that are weighted differently by the KL than the  $L_1$ . This is supported by the analysis of under-coverage and over-coverage. This situation happens more frequently under the likelihood strategy. As a consequence,

Dataset	Method	$L_1$ -ERT	KL-ERT	$KL_+$ -ERT	$KL_-$ -ERT
CIFAR10	cumulative	$0.072_{0.005}$	$-0.017_{0.008}$	$-0.030_{0.005}$	$0.012_{0.006}$
	likelihood	$0.016_{0.002}$	$0.028_{0.006}$	$0.007_{0.001}$	$0.022_{0.007}$
CIFAR100	cumulative	$0.041_{0.005}$	$0.191_{0.024}$	$0.016_{0.008}$	$0.175_{0.026}$
	likelihood	$0.007_{0.003}$	$0.409_{0.025}$	$0.085_{0.022}$	$0.323_{0.020}$
FashionMNIST	cumulative	$0.165_{0.004}$	$-0.260_{0.014}$	$-0.185_{0.010}$	$-0.075_{0.004}$
	likelihood	$0.098_{0.005}$	$-0.068_{0.008}$	$-0.042_{0.006}$	$-0.026_{0.005}$
MNIST	cumulative	$0.150_{0.006}$	$-0.216_{0.017}$	$-0.159_{0.012}$	$-0.057_{0.006}$
	likelihood	$0.145_{0.003}$	$-0.187_{0.007}$	$-0.128_{0.005}$	$-0.059_{0.002}$

Table 4: ERT scores obtained for the classification problems.

this strategy yields a larger value of  $KL_-$ -ERT than  $KL_+$ -ERT, since the KL divergence assigns more weight to such extreme situations.

## 5 Conclusions

Reliable estimation of conditional coverage is a key challenge for catalyzing further research progress in conformal prediction. We have addressed this challenge by moving from partition-based estimators to a functional-based framework. This shift provides a new perspective on how conditional coverage can be understood, measured, and improved. By framing the problem as one of risk minimization, we introduce a family of interpretable and reliable metrics that leverage the full expressive power of modern predictive models to detect and quantify conditional coverage violations.

Our framework unifies and extends existing diagnostics, transforming what was previously a collection of local, nonparametric estimators into a coherent, model-based approach. This transition not only provides more accurate and stable estimates but also offers a deeper understanding of what conditional coverage represents in practice. While the reliability of our metrics naturally depends on the quality of the learned classifier, this dependency is also a strength.

To encourage reproducibility and practical adoption, we release an open-source package implementing all proposed metrics alongside existing ones. Empirical results confirm the benefits of our approach and reveal that improving conditional coverage often leads to larger prediction sets. Understanding and controlling this balance is a promising direction for future research.

## Acknowledgements

Authors acknowledge funding from the European Union (ERC-2022-SYG-OCEAN-101071601). Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Research Council Executive Agency. Neither the European Union nor the granting authority can be held responsible for them. This publication is part of the Chair «Markets and Learning», supported by Air Liquide, BNP PARIBAS ASSET MANAGEMENT Europe, EDF, Orange and SNCF, sponsors of the Inria Foundation. This work has also received support from the French government, managed by the National Research Agency, under the France 2030 program with the reference «PR[AI]RIE-PSAI» (ANR-23-IACL-0008).

Finally, the authors would like to thank Eugène Berta for fruitful discussions regarding this work.

## References

Angelopoulos, A., S. Bates, J. Malik, and M. I. Jordan (2021). Uncertainty sets for image classifiers using conformal prediction. In *International Conference on Learning Representations*.

Angelopoulos, A. N. and S. Bates (2023). Conformal prediction: A gentle introduction. *Foundations and Trends in Machine Learning* 16(4), 494–591.

Bach, F. (2024). *Learning Theory from First Principles*. MIT Press.

Berta, E., D. Holzmüller, M. I. Jordan, and F. Bach (2025). Structured matrix scaling for multi-class calibration. *arXiv preprint arXiv:2511.03685*.

Braun, S., L. Aolaritei, M. I. Jordan, and F. Bach (2025). Minimum volume conformal sets for multivariate regression. *arXiv preprint arXiv:2503.19068*.

Braun, S., E. Berta, M. I. Jordan, and F. Bach (2025). Multivariate conformal prediction via conformalized gaussian scoring. *arXiv preprint arXiv:2507.20941*.

Bregman, L. M. (1967). The relaxation method of finding the common point of convex sets and its application to the solution of problems in convex programming. *USSR computational mathematics and mathematical physics* 7(3), 200–217.

Breiman, L. (2001). Random forests. *Machine learning* 45(1), 5–32.

Bröcker, J. (2009). Reliability, sufficiency, and the decomposition of proper scores. *Quarterly Journal of the Royal Meteorological Society: A Journal of the Atmospheric Sciences, Applied Meteorology and Physical Oceanography* 135(643), 1512–1519.

Cauchois, M., S. Gupta, and J. C. Duchi (2021). Knowing what you know: valid and validated confidence sets in multiclass and multilabel prediction. *Journal of Machine Learning Research* 22(81), 1–42.

Devroye, L., L. Györfi, and G. Lugosi (2013). *A Probabilistic Theory of Pattern Recognition*, Volume 31. Springer Science & Business Media.

Dheur, V., M. Fontana, Y. Estievenart, N. Desobry, and S. B. Taieb (2025). A unified comparative study with generalized conformity scores for multi-output conformal regression. In *International Conference on Machine Learning*.

Ding, T., A. Angelopoulos, S. Bates, M. Jordan, and R. J. Tibshirani (2023). Class-conditional conformal prediction with many classes. In *Advances in Neural Information Processing Systems*.

Ding, T., J.-B. Fermanian, and J. Salmon (2025). Conformal prediction for long-tailed classification. *arXiv preprint arXiv:2507.06867*.

Erickson, N., J. Mueller, A. Shirkov, H. Zhang, P. Larroy, M. Li, and A. Smola (2020). Autogluontabular: Robust and accurate automl for structured data. *arXiv preprint arXiv:2003.06505*.

Erickson, N., L. Purucker, A. Tschalzev, D. Holzmüller, P. M. Desai, D. Salinas, and F. Hutter (2025). Tabarena: A living benchmark for machine learning on tabular data. In *International Conference on Machine Learning*.

Feldman, S., S. Bates, and Y. Romano (2021). Improving conditional coverage via orthogonal quantile regression. In *Advances in Neural Information Processing Systems*.

Feldman, S., S. Bates, and Y. Romano (2023). Calibrated multiple-output quantile regression with representation learning. *Journal of Machine Learning Research* 24(24), 1–48.

Filliou, L., J. Silva-Rodríguez, I. B. Ayed, P.-H. Cournede, M. Vakalopoulou, S. Christodoulidis, and J. Dolz (2024). Are foundation models for computer vision good conformal predictors? *arXiv preprint arXiv:2412.06082*.

Foygel Barber, R., E. J. Candes, A. Ramdas, and R. J. Tibshirani (2021). The limits of distribution-free conditional predictive inference. *Information and Inference: A Journal of the IMA* 10(2), 455–482.

Gauthier, E., F. Bach, and M. I. Jordan (2025a). Adaptive coverage policies in conformal prediction. *arXiv preprint arXiv:2510.04318*.

Gauthier, E., F. Bach, and M. I. Jordan (2025b). Backward conformal prediction. In *Advances in Neural Information Processing Systems*.

Geurts, P., D. Ernst, and L. Wehenkel (2006). Extremely randomized trees. *Machine learning* 63(1), 3–42.

Gibbs, I., J. J. Cherian, and E. J. Candès (2025). Conformal prediction with conditional guarantees. *Journal of the Royal Statistical Society Series B: Statistical Methodology* 87, 1100–1126.

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

Gretton, A., O. Bousquet, A. Smola, and B. Schölkopf (2005). Measuring statistical dependence with hilbert-schmidt norms. In *International Conference on Algorithmic Learning Theory*.

Grinsztajn, L., K. Flöge, O. Key, F. Birkel, P. Jund, B. Roof, B. Jäger, D. Safaric, S. Alessi, A. Hayler, et al. (2025). TabPFN-2.5: Advancing the state of the art in tabular foundation models. *arXiv preprint arXiv:2511.08667*.

Guan, L. (2023). Localized conformal prediction: A generalized inference framework for conformal prediction. *Biometrika* 110(1), 33–50.

Györfi, L., M. Kohler, A. Krzyzak, and H. Walk (2005). *A Distribution-Free Theory of Nonparametric Regression*. Springer Science+Business Media.

Holzmüller, D., L. Grinsztajn, and I. Steinwart (2024). Better by default: Strong pre-tuned MLPs and boosted trees on tabular data. In *Advances in Neural Information Processing Systems*.

Izbicki, R., G. Shimizu, and R. B. Stern (2022). CD-split and HPD-split: Efficient conformal regions in high dimensions. *Journal of Machine Learning Research* 23(87), 1–32.

Jung, C., G. Noarov, R. Ramalingam, and A. Roth (2023). Batch multivalid conformal prediction. In *International Conference on Learning Representations*.

Kaur, J. N., M. I. Jordan, and A. Alaa (2025). Conformal prediction sets with improved conditional coverage using trust scores. *arXiv preprint arXiv:2501.10139*.

Ke, G., Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T.-Y. Liu (2017). Lightgbm: A highly efficient gradient boosting decision tree. *Advances in neural information processing systems* 30.

Lei, J., M. G'Sell, A. Rinaldo, R. J. Tibshirani, and L. Wasserman (2018). Distribution-free predictive inference for regression. *Journal of the American Statistical Association* 113(523), 1094–1111.

Lei, J. and L. Wasserman (2014). Distribution-free prediction bands for non-parametric regression. *Journal of the Royal Statistical Society Series B: Statistical Methodology* 76(1), 71–96.

Liu, S., J. Huang, and L. Ong (2025). Conformal prediction meets long-tail classification. *arXiv preprint arXiv:2508.11345*.

Messoudi, S., S. Destercke, and S. Rousseau (2021). Copula-based conformal prediction for multi-target regression. *Pattern Recognition* 120, 108101.

Messoudi, S., S. Destercke, and S. Rousseau (2022). Ellipsoidal conformal inference for multi-target regression. In *Conformal and Probabilistic Prediction with Applications*.

Mukama, B. C., S. Messoudi, S. Destercke, and S. Rousseau (2025). Copula-based conformal prediction for prioritized heterogeneous multi-task learning. *Pattern Recognition*, 112347.

Papadopoulos, H., K. Proedrou, V. Vovk, and A. Gammerman (2002). Inductive confidence machines for regression. In *European Conference on Machine Learning*.

Pedregosa, F., G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, et al. (2011). Scikit-learn: Machine learning in python. *Journal of Machine Learning Research* 12, 2825–2830.

Plassier, V., A. Fishkov, V. Dheur, M. Guizani, S. B. Taieb, M. Panov, and E. Moulines (2025). Rectifying conformity scores for better conditional coverage. In *International Conference on Machine Learning*.

Plassier, V., A. Fishkov, M. Guizani, M. Panov, and E. Moulines (2025). Probabilistic conformal prediction with approximate conditional validity. In *International Conference on Learning Representations*.

Plassier, V., A. Fishkov, M. Panov, and E. Moulines (2024). Conditionally valid probabilistic conformal prediction. In *arXiv preprint arXiv:2407.01794*.

Prokhorenkova, L., G. Gusev, A. Vorobev, A. V. Dorogush, and A. Gulin (2018). Catboost: unbiased boosting with categorical features. *Advances in neural information processing systems* 31.

Qu, J., D. Holzmüller, G. Varoquaux, and M. L. Morvan (2025). TabICL: A tabular foundation model for in-context learning on large data. In *International Conference on Machine Learning*.

Romano, Y., R. F. Barber, C. Sabatti, and E. J. Candès (2020). With malice towards none: Assessing uncertainty via equalized coverage. *Harvard Data Science Review*.

Romano, Y., E. Patterson, and E. Candes (2019). Conformalized quantile regression. In *Advances in Neural Information Processing Systems*.

Romano, Y., M. Sesia, and E. Candes (2020). Classification with valid and adaptive coverage. In *Advances in Neural Information Processing Systems*.

Sadinle, M., J. Lei, and L. Wasserman (2019). Least ambiguous set-valued classifiers with bounded error levels. *Journal of the American Statistical Association* 114(525), 223–234.

Shafer, G. and V. Vovk (2008). A tutorial on conformal prediction. *Journal of Machine Learning Research* 9(3), 371–421.

Sun, S. and R. Yu (2024). Copula conformal prediction for multi-step time series forecasting. In *International Conference on Learning Representations*.

Thurin, G., K. Nadjahi, and C. Boyer (2025). Optimal transport-based conformal prediction. In *International Conference on Machine Learning*.

Vapnik, V. (2000). *The Nature of Statistical Learning Theory*. Springer Science & Business Media.

Vovk, V. (2012). Conditional validity of inductive conformal predictors. In *Asian Conference on Machine Learning*.

Vovk, V., A. Gammerman, and G. Shafer (2005). *Algorithmic Learning in a Random World*. Springer.

Wang, F., L. Cheng, R. Guo, K. Liu, and P. S. Yu (2023). Equal opportunity of coverage in fair regression. In *Advances in Neural Information Processing Systems*.

Wang, Z., R. Gao, M. Yin, M. Zhou, and D. M. Blei (2023). Probabilistic conformal prediction using conditional random samples. In *International Conference on Artificial Intelligence and Statistics*.

Zhou, Y., L. Lindemann, and M. Sesia (2024). Conformalized adaptive forecasting of heterogeneous trajectories. In *International Conference on Machine Learning*.

Zhu, Y., D. Hernández, Y. He, Z. Ding, B. Xiong, E. Kharlamov, and S. Staab (2025). Predicate-conditional conformalized answer sets for knowledge graph embeddings. *arXiv preprint arXiv:2505.16877*.

# Appendix

## A Additional metrics

**Group-based diagnostics.** We start by reviewing additional group-based diagnostics.

- **Equalized / group-wise coverage.** A fairness-style requirement is that every group attains the nominal coverage:

$$\mathbb{P}_{Y|g(X)=\mathbf{g}}(Y \in C_\alpha(X) | g(X) = \mathbf{g}) = 1 - \alpha \quad \forall \mathbf{g} \in \mathcal{G}.$$

This notion appears in the conformal fairness literature (e.g, [Romano et al. 2020](#); [Ding et al. 2025](#)) and is evaluated in practice by returning the values  $C_{\mathbf{g}}$  for all  $\mathbf{g}$ .

- **Feature-stratified coverage (FSC).** To focus on the worst-off group, FSC reports the minimal empirical coverage across groups:

$$\text{FSC} = \min_{\mathbf{g} \in \mathcal{G}} C_{\mathbf{g}}.$$

FSC highlights subgroups where coverage is lowest and has been used in several works (e.g, [Angelopoulos and Bates 2023](#); [Ding et al. 2023](#); [Jung et al. 2023](#)). This metric is often viewed as a fairness measure, as it focuses on the group that exhibits the poorest coverage.

All of the above group-based diagnostics are sensitive to how the groups  $\mathcal{G}$  are chosen. Most of the time, they are created by partitioning the feature space, either with clustering methods, or by using categorical features. Much of the recent work focuses on finding or learning useful partitions that reveal conditional coverage violations, by applying the induced CovGap or FSC metric. In the following, unless otherwise specified, the groups used to evaluate FSC and CovGap are obtained by clustering the feature space. We will use the following notation to refer to alternative grouping strategies.

- **Equal opportunity of coverage (EOC).** To account for differences in outcomes, EOC requires that coverage rates across protected groups are equal conditional on the true label; in regression this means that for each outcome value  $y$  (or a discretization of  $y$ ), the coverage within each protected subgroup should match. This idea was introduced by [Wang et al. \(2023\)](#). However, defining groups based on the outcome can lead to misleading metrics. For instance, consider an interval predictor  $[q_{\alpha/2}, q_{1-\alpha/2}]$  for  $Y \sim \mathcal{N}(0, 1)$ , where  $q_{\alpha/2}$  and  $q_{1-\alpha/2}$  are the  $\alpha/2$  and  $1 - \alpha/2$  quantiles of the normal distribution respectively. If all extreme values of  $Y$  are grouped together, the resulting group may show a coverage of zero, even though the prediction interval is conditionally valid.
- **Size-stratified coverage (SSC).** Instead of grouping by features, SSC groups examples by the size (e.g., volume or cardinality) of their prediction sets. It is model-agnostic and useful when  $X$  is high-dimensional because it avoids requiring semantically meaningful groups. However, SSC can fail to detect conditional-coverage problems in some settings. Consider, for example regression with the nonconformity score  $S(X, Y) := |Y - f(X)|$  (cf. [Angelopoulos et al., 2021](#)). This score cannot capture heteroskedasticity ([Vovk, 2012](#)), outputting prediction sets with the same sizes independently of the covariate. Thus, in this case, SSC will not reveal coverage failures. Furthermore, it requires access to the prediction set sizes which can be computationally costly for some strategies.

**Representation-based diagnostics (dependence on auxiliary variables).** An alternative to grouping is to measure statistical dependence between the coverage indicator

$$Z := \mathbb{1}\{Y \in C_\alpha(X)\}$$

and auxiliary variables  $V$  (for example, prediction-set size, nonconformity score, residuals, or other model-derived quantities). If  $Z$  is independent of  $V$ , this is evidence that coverage does not systematically vary with  $Z$ . [Feldman et al. \(2021\)](#) proposed two measures induced by this remark.

- **Pearson’s correlation.** This is a simple measure of linear dependence,

$$R_{\text{corr}}(Z, V) = \frac{\text{Cov}(Z, V)}{\sqrt{\text{Var}(Z) \text{Var}(V)}},$$

where  $V$  is the prediction-set size. This is fast and interpretable but only captures linear relationships.

- **HSIC (Hilbert–Schmidt independence criterion).** The HSIC is a nonparametric kernel-based dependence measure that can detect arbitrary nonlinear dependence. Given a suitable pair of kernels, HSIC estimates the maximum mean discrepancy (MMD) (Gretton et al., 2005) between the joint distribution  $\mathbb{P}_{Z,V}$  and the product of marginals  $\mathbb{P}_Z \otimes \mathbb{P}_V$ . Feldman et al. (2021) defined the metric based on HSIC:

$$R_{\text{HSIC}}(Z, V) = \sqrt{\text{HSIC}(Z, V)},$$

where the square root emphasizes small deviations from independence and gives a loss that is easier to interpret and optimize. Here,  $V$  is again the prediction-set size.

## B Estimating Bregman divergences with ERTs

The following proposition is related to Theorem 3.1 and studies the setting when there is a single proper score  $\ell$  that estimates a distance  $d(1 - \alpha, p)$  simultaneously for all  $\alpha$  and  $p$ .

**Proposition 5.1.** *A function  $d(p, q)$  arises as the divergence of some proper scoring rule if and only if there exists a convex function  $\varphi : [0, 1] \rightarrow \mathbb{R}$  such that*

$$d(p, q) = D_\varphi(q \| p) = \varphi(q) - \varphi(p) - (q - p) s(p)$$

for any choice of subgradient  $s(p) \in \partial\varphi(p)$ . Furthermore, the associated proper score is defined as

$$\ell_\varphi(p, y) := \varphi(p) + (y - p) s(p).$$

and

$$\ell_\varphi\text{-ERT} = \mathbb{E}_X[D_\varphi(p(X) \| 1 - \alpha)].$$

*Proof.* Let  $\ell$  be a proper scoring rule for binary outcomes and write  $p, q \in [0, 1]$  for probabilities of the event  $Y = 1$ . Define

$$\varphi(q) := \mathbb{E}_{Y \sim q}[\ell(q, Y)] = q \ell(q, 1) + (1 - q) \ell(q, 0).$$

Properness of  $\ell$  means that for every fixed  $p \in [0, 1]$  and every  $q \in [0, 1]$ ,

$$\varphi(q) = \mathbb{E}_{Y \sim q}[\ell(q, Y)] \leq \mathbb{E}_{Y \sim q}[\ell(p, Y)] = q \ell(p, 1) + (1 - q) \ell(p, 0).$$

Rearranging gives

$$\varphi(q) \leq \varphi(p) + (q - p)(\ell(p, 1) - \ell(p, 0))$$

so  $\varphi$  is convex on  $[0, 1]$  and the function  $p \mapsto \ell(p, 1) - \ell(p, 0)$  defines a subgradient of  $\varphi$  at  $p$ . Denote by  $s(p) \in \partial\varphi(p)$  any such subgradient. Then for every  $p, q \in [0, 1]$ ,

$$\mathbb{E}_q[\ell(p, Y)] = \varphi(p) + (q - p) s(p),$$

and hence the divergence of  $\ell$  satisfies

$$d_\ell(p, q) = \mathbb{E}_q[\ell(p, Y)] - \mathbb{E}_q[\ell(q, Y)] = \varphi(p) + (q - p) s(p) - \varphi(q) = \varphi(q) - \varphi(p) - (q - p) s(p),$$

which is exactly the Bregman divergence  $D_\varphi(q \| p)$  generated by  $\varphi$ .

Conversely, let  $\varphi : [0, 1] \rightarrow \mathbb{R}$  be convex and for each  $p \in [0, 1]$  choose a subgradient  $s(p) \in \partial\varphi(p)$ . Define a score  $\ell_\varphi$  by

$$\ell_\varphi(p, y) := \varphi(p) + (y - p) s(p).$$

by construction, and convexity of  $\varphi$  implies for every  $p$  and  $q$ ,

$$\mathbb{E}_q[\ell_\varphi(p, Y)] = \varphi(p) + (q - p)s(p) \geq \varphi(q).$$

Thus  $\ell_\varphi$  is proper, and its divergence equals

$$\mathbb{E}_q[\ell_\varphi(p, Y)] - \mathbb{E}_q[\ell_\varphi(q, Y)] = \varphi(p) + (q - p)s(p) - \varphi(q) = D_\varphi(q\|p).$$

Therefore a function  $d(p, q)$  arises as the divergence of a proper scoring rule if and only if it is the Bregman divergence of some convex generator  $\varphi$ , as claimed. Note that the score  $\ell_\varphi$  is determined up to addition of an arbitrary function of the outcome whose expectation under every  $q$  is zero, and that when  $\varphi$  is differentiable the subgradient  $s(p)$  may be replaced by the derivative  $\varphi'(p)$ .

Using this proper score to evaluate the  $\ell_\varphi$ -ERT we get

$$\begin{aligned} \ell_\varphi\text{-ERT} &= \mathcal{R}_{\ell_\varphi}(1 - \alpha) - \mathcal{R}_{\ell_\varphi}(p) \\ &= \mathbb{E}_{X, Z}[\ell_\varphi(1 - \alpha, Z) - \ell_\varphi(p(X), Z)] \\ &= \mathbb{E}_X[\mathbb{E}_Z[\ell_\varphi(1 - \alpha, Z) - \ell_\varphi(p(X), Z)|X]] \\ &= \mathbb{E}_X[\mathbb{E}_{Z \sim p(X)}[\ell_\varphi(1 - \alpha, Z) - \ell_\varphi(p(X), Z)]] \\ &= \mathbb{E}_X[d_{\ell_\varphi}(1 - \alpha, p(X))] \\ &= \mathbb{E}_X[D_\varphi(p(X)\|1 - \alpha)]. \end{aligned} \quad \square$$

## C Link with Gibbs et al. (2025)

To get predictive sets with conditional guarantees, [Gibbs et al. \(2025\)](#) used the fact that conditional coverage holds if for all measurable function  $\varphi$ ,

$$\mathbb{E}_{X, Z}[\varphi(X)(1 - \alpha - Z)] = 0.$$

To obtain a more operational interpretation, suppose we restrict attention to predictors of the form  $h(x) = \theta^T \phi(x)$ , where  $\phi(x)$  is a feature map. The associated mean squared risk is

$$F(\theta) = \mathbb{E}_{X, Z}[(\theta^T \phi(X) - Z)^2]$$

and has gradient

$$\nabla_\theta F(\theta) = 2 \mathbb{E}_{X, Z}[(\theta^T \phi(X) - Z)\phi(X)].$$

Write  $\theta_\alpha$  such that  $h_{\theta_\alpha}(X) = 1 - \alpha$ , (that exists when  $\phi(X)$  has a non-zero constant component).

For this class of models, our metric  $L_2$ -ERT compares any  $\theta$  to this target via

$$F(\theta_\alpha) - \inf_\theta F(\theta).$$

When conditional coverage holds, both the gradient and the  $L_2$ -ERT are equal to zero. In this sense, the quantity  $F(\theta_\alpha) - F(\theta)$  offers more interpretability than the gradient alone. While the gradient describes only a local direction in the parameter space of improvement, the risk difference has a clear interpretation that has already been discussed. Indeed, when  $F(\theta_\alpha) - \inf_\theta F(\theta) = 0$  we have  $\mathbb{E}_{X, Z}[(1 - \alpha - Z)\phi(X)] = 0$  but it can be that  $F(\theta_\alpha) - \inf_\theta F(\theta)$  is very small but that  $\|\mathbb{E}_{X, Z}[(1 - \alpha - Z)\phi(X)]\|_2$  is very large. That is why  $\mathbb{E}_{X, Z}[\varphi(X)(1 - \alpha - Z)]$  cannot be used as an interpretable quantity to quantify conditional miscoverage deviation, but only to assess if there is conditional coverage or not.

## D Extension to a conditional coverage rule

While conditional coverage is often defined with a fixed target level,

$$\mathbb{P}(Y \in C_\alpha(X_{\text{test}}) \mid X_{\text{test}}) = 1 - \alpha \quad \mathbb{P}_X\text{-almost surely},$$

ensuring that the prediction set  $C_\alpha(X)$  covers the true label with probability  $1 - \alpha$  for every possible input  $X$ , a uniform coverage level may be neither necessary nor desirable in practice. In many settings,

one may wish to adapt the coverage level to reflect varying uncertainty, heteroskedastic noise, or task-specific risk preferences. Recent conformal prediction methods allow adapting  $\alpha$  dynamically to optimize other objectives (Gauthier et al., 2025b,a).

To formalize this flexibility, we introduce a *conditional miscoverage rule*  $\alpha : \mathcal{X} \rightarrow [0, 1]$ , which prescribes a desired miscoverage level  $\alpha(X)$  that may vary with the input features. Under this generalized framework, the target conditional coverage condition becomes

$$\mathbb{E}[\mathbb{1}\{Y \in C_\alpha(X_{\text{test}})\} \mid X_{\text{test}}] = 1 - \alpha(X_{\text{test}}) \quad \mathbb{P}_X\text{-almost surely.}$$

This formulation recovers the standard conformal setting in the special case where  $\alpha(X)$  is constant, while enabling the analysis of predictors designed to achieve non-uniform, data-dependent coverage guarantees. It thus provides a principled way to study how prediction methods align with arbitrary, application-driven notions of conditional reliability. This could be useful for example in classification, where the discrete nature of the target may not allow to achieve constant coverage.

We next adapt our framework for the  $\ell$ -ERT metric under this setting by writing

$$\ell\text{-ERT} = \mathcal{R}_\ell(1 - \alpha) - \mathcal{R}_\ell(p) = \mathbb{E}\left[(d_\ell(1 - \alpha(X), p(X))\right].$$

and its variational form

$$\ell\text{-ERT}(h) = \mathcal{R}_\ell(1 - \alpha) - \mathcal{R}_\ell(h) = \mathbb{E}_{X,Y}\left[\ell(1 - \alpha(X), Y) - \ell(h(X), Y)\right].$$

The formulas above work for the  $L_2$ -ERT and the KL-ERT, where the proper score  $\ell$  does not depend on  $1 - \alpha$ . If general distance metrics should be estimated as in Theorem 3.1, such as for the  $L_1$ -ERT, the proper scoring rule  $\ell$  needs to depend on  $\alpha(X)$ , and we obtain

$$\ell\text{-ERT} = \mathbb{E}\left[(d_{\ell_{\alpha(X)}}(1 - \alpha(X), p(X))\right].$$

and its variational form

$$\ell\text{-ERT}(h) = \mathbb{E}_{X,Y}\left[\ell_{\alpha(X)}(1 - \alpha(X), Y) - \ell_{\alpha(X)}(h(X), Y)\right].$$

The remaining components of the procedure remain unchanged. For convenience, we refer to the resulting metrics using the same terminology as in the fixed  $1 - \alpha$  case, since that setting corresponds to a particular instance of this more general framework.

## E Benchmarking strategies

The strategies we compare can be differentiated in four main groups—ones that uses density estimation, latent spaces, hyper-rectangles, or minimizing a quantity while ensuring marginal coverage. We build upon the work of Dheur et al. (2025) that already explained most of those strategies, but we recall their specificities here for completeness.

Among the density-based methods, given a predictive density  $\hat{p}(y|x)$ , the benchmarked strategies are:

- **DR-CP** (Sadinle et al., 2019): Defines the conformity score as  $S_{\text{DR-CP}}(X, Y) = -\hat{p}(Y|X)$ , leading to prediction regions that are density superlevel sets,  $C_{\text{DR-CP}}(X) = \{y : \hat{p}(y|X) \geq -\hat{q}\}$ .
- **C-HDR** (Izbicki et al., 2022): Conformalize the highest predictive density (HPD) by using the nonconformity score  $S_{\text{HDP}}(X, Y) = \mathbb{P}_{y \sim \hat{p}(\cdot|X)}(\hat{p}(y|X) \geq \hat{p}(Y|X))$ . It then produces regions  $C_{\text{C-HDR}}(X) = \{y : \hat{f}(y|X) \geq \hat{t}_q\}$ , where  $\hat{t}_q$  defines the highest density region (HDR) at level  $\hat{q}$ .
- **PCP** (Wang et al., 2023): Draws  $L$  samples  $\tilde{Y}^{(l)} \sim \hat{p}_{Y|x}$  with  $\hat{p}_{Y|x}$  the estimated conditional distribution, and defines conformity as the distance to the nearest sample,  $S_{\text{PCP}}(X, Y) = \min_{l \in [L]} \|Y - \tilde{Y}^{(l)}\|_2$ ; the corresponding region is a union of  $L$  balls centered at the sampled points.
- **HD-PCP** (Wang et al., 2023): Extends PCP by retaining only the top  $\lfloor (1 - \alpha)L \rfloor$  samples with highest density, concentrating the prediction region on high-density areas.

- **C-PCP** (Dheur et al., 2025): Estimates the conditional CDF of the conformity score  $S(X, Y)$ ,

$$S_{\text{CDF}}(x, y) = \mathbb{P}(S_W(X, Y) \leq S_W(x, y) \mid X = x),$$

using a Monte Carlo approximation with  $K$  samples

$$S_{\text{ECDF}}(x, y) = \frac{1}{K} \sum_{k=1}^K \mathbb{1}[S_W(x, \hat{Y}^{(k)}) \leq S_W(x, y)], \quad \hat{Y}^{(k)} \sim \hat{F}_{Y|x}.$$

When  $S(x, y) = S_{\text{PCP}}(x, y)$ , this yields

$$S_{\text{C-PCP}}(x, y) = \frac{1}{K} \sum_{k \in [K]} \mathbb{1} \left\{ \min_{l \in [L]} \|\hat{Y}^{(k)} - \tilde{Y}^{(l)}\|_2 \leq \min_{l \in [L]} \|y - \tilde{Y}^{(l)}\|_2 \right\}.$$

- **CP2-PCP** (Plassier et al., 2024): Builds predictive sets by using samples from an implicit conditional generative model. For each calibration point it uses two independent draws from the conditional generator to define a conformity score and an inflation parameter  $\tau$  that accounts for the conditional mass around likely outputs. At prediction time it forms a union of balls around new generated samples, with their size chosen to guarantee marginal validity while improving approximate conditional adaptivity.
- **MSE**: Naïve multivariate generalization of the *univariate* score  $S(X, Y) = |Y - f(X)|$ , by using the *multivariate* score  $S(X, Y) = \|Y - f(X)\|_2$ .

Among the latent space-based methods, the benchmarked strategies are:

- **STDQR** (Feldman et al., 2023): Constructs multivariate prediction regions in a latent space  $\mathcal{Z}$  to overcome limitations of standard multivariate prediction methods. Instead of using directional quantile regression as originally introduced, we follow Dheur et al. (2025) procedure where the region  $R_{\mathcal{Z}}$  with coverage  $1 - \alpha$  is constructed by selecting the  $1 - \alpha$  proportion of latent samples closest to the origin, ensuring correct coverage directly in the latent space. These latent regions are then mapped to the output space  $\mathcal{Y}$  via a conditional generative model (originally a CVAE, here replaced with a normalizing flow). A conformalization step refines coverage by creating a grid of latent samples, mapping them to  $\mathcal{Y}$ , and forming small balls around each mapped point.
- **L-CP** (Dheur et al., 2025): Defines conformity in a latent space using an invertible conditional generative model  $\hat{Q} : \mathcal{Z} \times \mathcal{X} \rightarrow \mathcal{Y}$ . A latent variable  $Z \sim \mathcal{N}(0, I_d)$  is mapped to the output space via  $\hat{Q}$ , and the conformity score is measured in latent space as

$$S_{\text{L-CP}}(X, Y) = \|\hat{Q}^{-1}(Y; X)\|.$$

The prediction region is obtained by taking a ball of radius  $\hat{q}$  around the origin in latent space and mapping it back to the output space. This method avoids grid-based directional quantile regression, improving scalability and computational efficiency, and generalizes distributional conformal prediction to multivariate outputs.

Among the hyper-rectangle-based methods, the benchmarked strategies are:

- **CopulaCPTS** (Sun and Yu, 2024): This method models the joint dependence between marginal conformity scores via a copula. The calibration data are split into two sets:  $\mathcal{D}_{\text{cal-1}}$  to estimate empirical CDFs  $\hat{F}_i$  of conformity scores for each output dimension  $i \in [d]$ , and  $\mathcal{D}_{\text{cal-2}}$  to calibrate the copula parameters. The optimal thresholds  $s_1^*, \dots, s_d^*$  are obtained by minimizing a coverage-based loss, ensuring marginal validity while reducing region size. The final prediction region is

$$C_{\text{CopulaCPTS}}(X) = \{y \in \mathcal{Y} : S_i(X, y_i) < s_i^*, \forall i \in [d]\}.$$

Other copula's based strategies include Messoudi et al. (2021); Mukama et al. (2025).

- **HR** (Romano et al., 2019; Zhou et al., 2024): Constructs axis-aligned (hyper-rectangular) prediction regions by fitting univariate quantiles  $\tilde{q}_{\alpha/2}(x)_i$  and  $\tilde{q}_{1-\alpha/2}(x)_i$  following Romano et al. (2019) for each output dimension  $i \in [k]$ , with  $\tilde{\alpha} = 2(1 - (1 - \alpha)^{1/k})$ . The conformity score is defined as (inspired from Zhou et al. (2024) which extends uni-variate scorings to multi-variate ones)

$$S_{\text{HR}}(X, Y) = \max_{i \in [k]} \left\{ \tilde{q}_{\alpha/2}(X)_i - Y_i, Y_i - \tilde{q}_{1-\alpha/2}(X)_i \right\},$$

yielding rectangular prediction regions aligned with coordinate axes.

Finally, among the strategies which minimizes the size of the prediction sets, we use:

- **MVCS** (Braun et al., 2025): Minimizes the volume of the sets  $\{y \in \mathbb{R}^k, \|M(X)(y - f(X))\|_p \leq 1\}$  where  $M(X)$  is positive definite,  $f(X) \in \mathbb{R}^k$ , and  $p > 0$  defines a  $p$ -norm, while ensuring valid marginal coverage. The conformalization set is done with the score  $S(X, Y) = \|M(X)(Y - f(X))\|_p$ .

## F Strong classifiers

Here, we provide more details on the classifiers used in Section 4. In the following, we will refer to training and test data for the data that the classifier  $h$  is trained and evaluated on, not to be confused with the data that the original prediction set method  $C_\alpha$  is trained on. We employ the following classifiers in our evaluation:

**Tabular foundation models.** We evaluate the recent models **RealTabPFN-2.5** (Grinsztajn et al., 2025) and **TabICLv1.1** (Qu et al., 2025). These models can predict  $h(x_1^{\text{test}}), \dots, h(x_n^{\text{test}})$  with a single forward pass through a neural network that takes both the test input and the entire training set into account. They have been found to perform very well already without hyperparameter tuning (Erickson et al., 2025).

**Gradient-boosted decision trees.** We choose two representatives: CatBoost (Prokhorenkova et al., 2018) is known for its strong default performance. We adopt its hyperparameters from AutoGluon (Erickson et al., 2020) and TabArena (Erickson et al., 2025), using 300 early stopping rounds instead of AutoGluon’s custom early stopping logic. To obtain a faster model, we use LightGBM (Ke et al., 2017) with cheaper hyperparameters adapted from the tuned defaults of Holzmüller et al. (2024), reducing the number of early stopping rounds to 100 and using cross-entropy loss for early stopping (as for CatBoost). Both CatBoost and LightGBM are fitted in parallel for eight inner cross-validation folds for each outer cross-validation fold, following TabArena. The inner cross-validation folds are used for early stopping, and the final validation predictions are concatenated and used to fit a quadratic scaling post-hoc calibrator (Berta et al., 2025). Test set predictions are made based on the post-hoc calibrator applied to the average of the eight models’ predictions.

**Bagging models.** Random Forest (Breiman, 2001) is a popular baseline for tabular ML. Extremely randomized trees (ExtraTrees/XT, Geurts et al., 2006) are a more randomized variant that performs similarly while being faster (Erickson et al., 2025). We fit both models with 300 estimators and otherwise use the default hyperparameters from `scikit-learn` (Pedregosa et al., 2011).

**PartitionWise** Partition-wise estimation first groups samples in the feature space and then predicts by using the average label within each group. The predictor relies on KMeans to form the partitions, and the number of clusters is selected through a fourth root rule based on the test set size, which keeps the model flexible while avoiding clusters that are too small. At inference time, each new sample is assigned to its nearest cluster and the model predicts the mean stored for that cluster.

**Trade-offs.** We chose to only fit boosted trees with inner cross-validation, both because they need validation sets for early stopping and because they are still reasonably fast with parallelization. The use of cross-validation also allows for the application of post-hoc calibration. Other methods might also benefit from post-hoc calibration, especially for non-L1 metrics, at the cost of higher runtime due to cross-validation.

**Discussion and other options.** We omit from-scratch trained tabular neural networks from our comparison as they are relatively slow, especially on CPUs, and their un-tuned performance is sub-optimal (Erickson et al., 2025). If runtime is less of a concern, for ideal sample-efficiency, automated machine learning methods such as AutoGluon (Erickson et al., 2020) can be employed that combine multiple models and hyperparameter setting, ensembling, and post-hoc calibration. However, when these methods are used with time limits, the result may not be reproducible.

**Hardware.** We ran tabular foundation models on GPUs (NVIDIA RTX8000, V100 and RTX6000) and the other models on CPUs (Cascade Lake Intel Xeon 5217 with 8 cores and AMD EPYC 7302 with 16 cores).

## G Additional information regarding the experiments

### G.1 Details on the datasets

See Table 5 for details on the datasets used for the classifiers comparison, Table 6 for details on the uni-variate datasets and Table 7 for the multivariate ones.

Dataset	Number of samples	Number of test samples	Number of features
physicochemical_protein	45730	22865	9
Food_Delivery_Time	45593	22797	10
diamonds	53940	26970	9
superconductivity	21263	10632	81

Table 5: Description of the univariate datasets.

Dataset	Number of samples	Number of test samples	Number of features
airlines	13750	1375	40
bank8FM	8192	820	8
cpu-act	8192	820	21
house-8L	22784	2280	8
miami	13932	1394	16
sulfur	10081	1009	6

Table 6: Description of the univariate datasets.

Dataset	Number of samples	Number of test samples	Number of features	Dimension of targets
Bias	7752	776	22	2
CASP	45730	4574	8	2
House	21613	2162	17	2
rf1	9125	914	64	8
rf2	9125	914	576	8
Taxi	61286	6130	6	2

Table 7: Description of the multivariate datasets.

## H Additional experiments

**Synthetic dataset** See Figures 9 & 10 & 11 & 12.

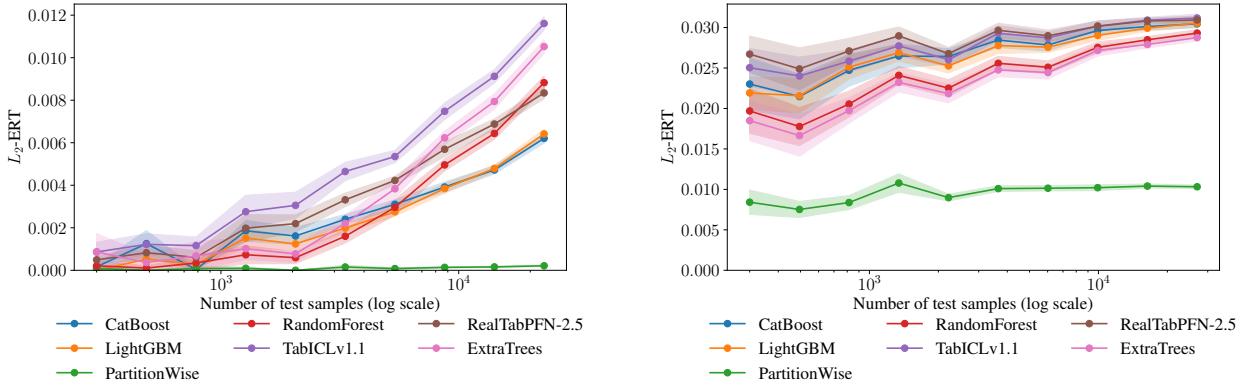


Figure 9: Illustration of the estimation of  $L_2$ -ERT for different classifiers as a number of sampled data available. **Left:** physicochemical\_protein dataset **Right:** Diamonds dataset

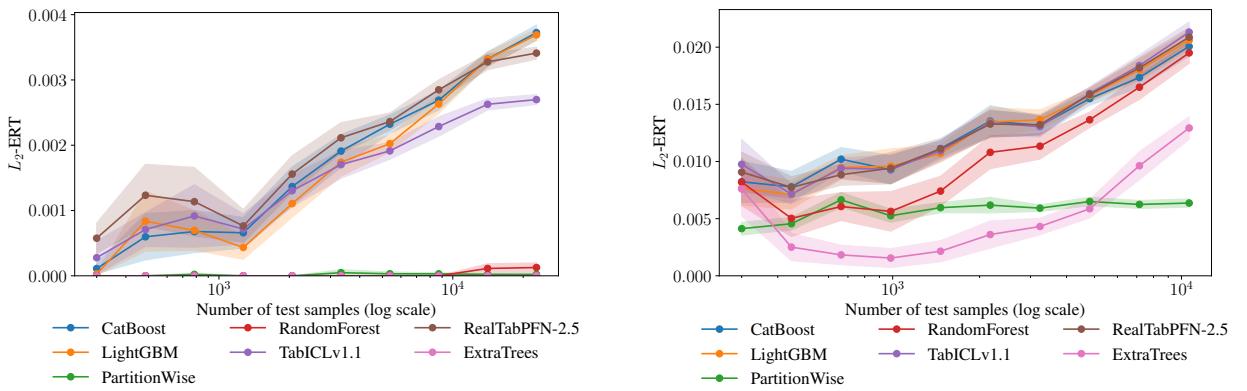


Figure 10: Illustration of the estimation of  $L_2$ -ERT for different classifiers as a number of sampled data available. **Left:** Food\_Delivery\_Time dataset **Right:** Superconductivity dataset

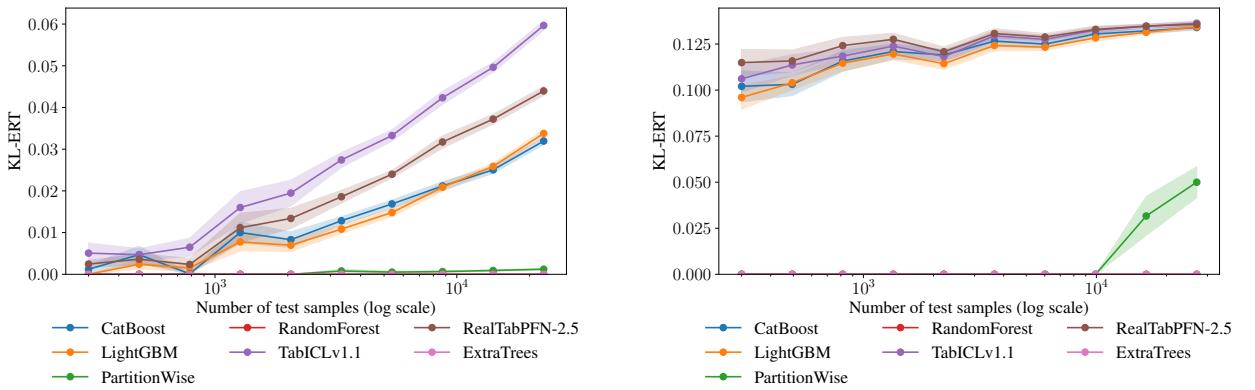


Figure 11: Illustration of the estimation of KL-ERT for different classifiers as a number of sampled data available. **Left:** physicochemical\_protein dataset **Right:** Diamonds dataset

**Uni-variate regression.** See Figures 13 & 14 & 15.

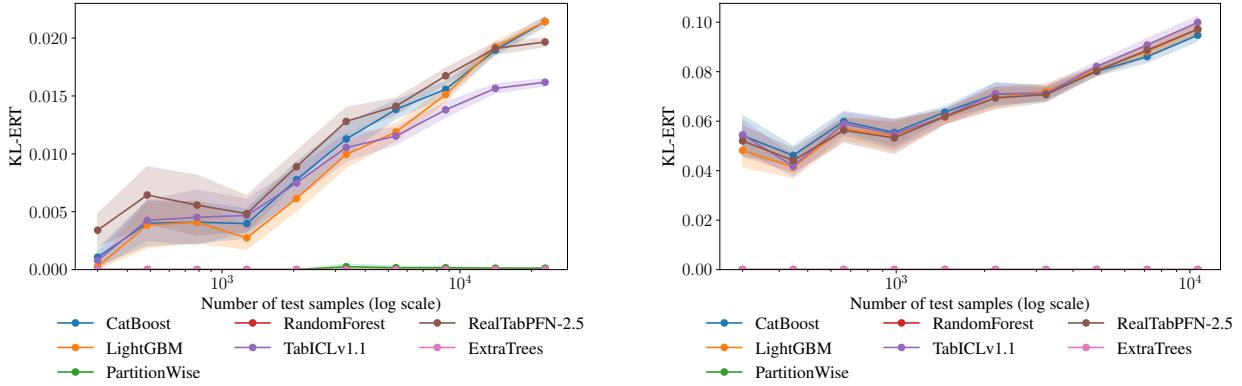


Figure 12: Illustration of the estimation of KL-ERT for different classifiers as a number of sampled data available. **Left:** Food\_Delivery\_Time dataset **Right:** Superconductivity dataset

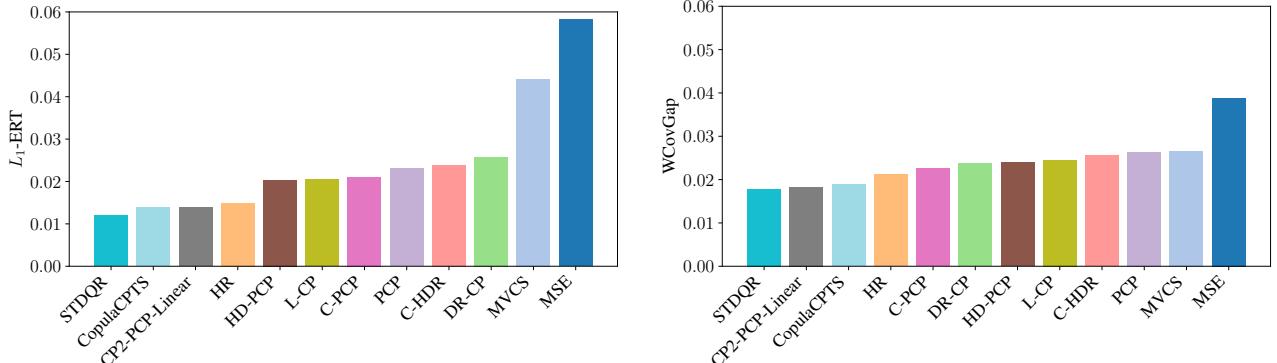


Figure 13: Metric values averaged across all datasets for all methods in univariate regression. **Left:**  $L_1$ -ERT (lower is better). **Right:** WCovGap (lower is better)

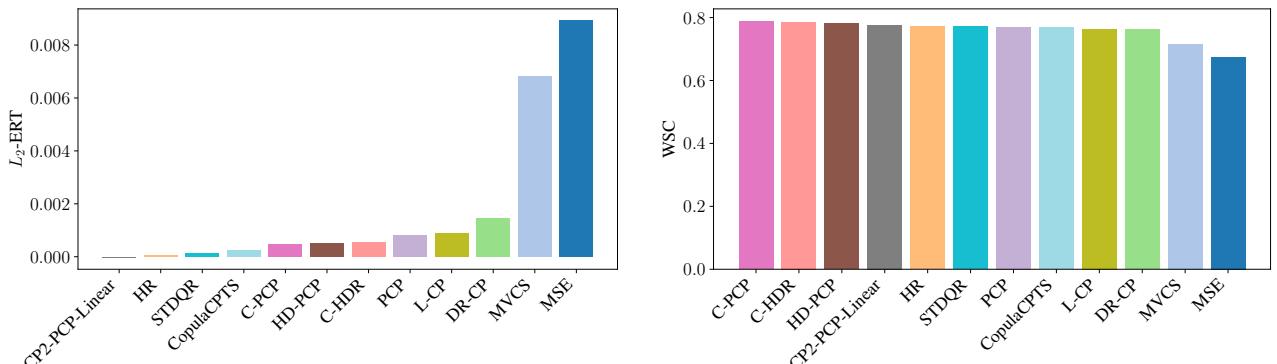


Figure 14: Metric values averaged across all datasets for all methods in univariate regression. **Left:**  $L_2$ -ERT (lower is better). **Right:** WSC (closer to 0.9 is better).

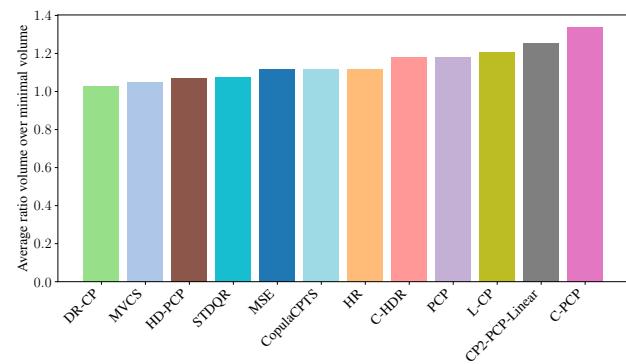


Figure 15: Normalized set sizes averaged all datasets in univariate regression, where the normalization is done by dividing each volume by the smallest volume across all methods (smaller is better).

Dataset	Method	$L_1$ -ERT	$L_2$ -ERT	WSC	WCovGap	Size
Allerons	MSE	0.0287 <sub>0.0058</sub>	0.0010 <sub>0.0004</sub>	0.777 <sub>0.015</sub>	0.025 <sub>0.004</sub>	<b>1.42<sub>0.03</sub></b>
	MVCS	0.0249 <sub>0.0042</sub>	0.0007 <sub>0.0004</sub>	0.768 <sub>0.019</sub>	0.023 <sub>0.003</sub>	1.43 <sub>0.02</sub>
	HR	0.0126 <sub>0.0000</sub>	−0.0004 <sub>0.0000</sub>	0.804 <sub>0.000</sub>	0.017 <sub>0.000</sub>	1.46 <sub>0.00</sub>
	DR-CP	0.0024 <sub>0.0000</sub>	−0.0007 <sub>0.0000</sub>	0.783 <sub>0.000</sub>	<b>0.009<sub>0.000</sub></b>	1.46 <sub>0.00</sub>
	C-HDR	0.0293 <sub>0.0000</sub>	0.0013 <sub>0.0000</sub>	0.783 <sub>0.000</sub>	0.026 <sub>0.000</sub>	1.64 <sub>0.00</sub>
	PCP	0.0072 <sub>0.0000</sub>	−0.0000 <sub>0.0000</sub>	0.779 <sub>0.000</sub>	0.020 <sub>0.000</sub>	1.58 <sub>0.00</sub>
	HD-PCP	0.0004 <sub>0.0000</sub>	−0.0005 <sub>0.0000</sub>	0.797 <sub>0.000</sub>	0.027 <sub>0.000</sub>	1.44 <sub>0.00</sub>
	C-PCP	0.0393 <sub>0.0000</sub>	0.0020 <sub>0.0000</sub>	<b>0.804<sub>0.000</sub></b>	0.026 <sub>0.000</sub>	1.80 <sub>0.00</sub>
	CP2-PCP	0.0052 <sub>0.0000</sub>	−0.0003 <sub>0.0000</sub>	0.799 <sub>0.000</sub>	0.022 <sub>0.000</sub>	1.69 <sub>0.00</sub>
	L-CP	0.0132 <sub>0.0000</sub>	−0.0001 <sub>0.0000</sub>	0.779 <sub>0.000</sub>	0.020 <sub>0.000</sub>	1.55 <sub>0.00</sub>
	STDQR	<b>−0.0018<sub>0.0000</sub></b>	−0.0005 <sub>0.0000</sub>	0.790 <sub>0.000</sub>	<u>0.015<sub>0.000</sub></u>	1.44 <sub>0.00</sub>
	CopulaCPTS	<u>−0.0004<sub>0.0000</sub></u>	<b>−0.0009<sub>0.0000</sub></b>	0.794 <sub>0.000</sub>	0.022 <sub>0.000</sub>	1.44 <sub>0.00</sub>
MiamiHousing2016	MSE	0.0736 <sub>0.0050</sub>	0.0088 <sub>0.0015</sub>	0.647 <sub>0.014</sub>	0.056 <sub>0.012</sub>	0.91 <sub>0.02</sub>
	MVCS	0.0509 <sub>0.0078</sub>	0.0047 <sub>0.0013</sub>	0.713 <sub>0.023</sub>	0.040 <sub>0.006</sub>	<b>0.82<sub>0.02</sub></b>
	HR	0.0274 <sub>0.0001</sub>	0.0005 <sub>0.0002</sub>	0.771 <sub>0.001</sub>	0.028 <sub>0.001</sub>	0.87 <sub>0.00</sub>
	DR-CP	0.0308 <sub>0.0003</sub>	0.0010 <sub>0.0002</sub>	0.746 <sub>0.000</sub>	0.033 <sub>0.001</sub>	0.85 <sub>0.00</sub>
	C-HDR	0.0157 <sub>0.0003</sub>	0.0002 <sub>0.0003</sub>	<u>0.800<sub>0.003</sub></u>	0.018 <sub>0.000</sub>	0.95 <sub>0.00</sub>
	PCP	0.0277 <sub>0.0005</sub>	0.0011 <sub>0.0005</sub>	0.766 <sub>0.003</sub>	0.022 <sub>0.000</sub>	0.96 <sub>0.01</sub>
	HD-PCP	0.0324 <sub>0.0069</sub>	0.0016 <sub>0.0004</sub>	0.757 <sub>0.000</sub>	0.021 <sub>0.001</sub>	0.86 <sub>0.00</sub>
	C-PCP	0.0184 <sub>0.0026</sub>	<u>0.0002<sub>0.0003</sub></u>	<b>0.802<sub>0.002</sub></b>	0.023 <sub>0.002</sub>	1.06 <sub>0.00</sub>
	CP2-PCP	0.0294 <sub>0.0040</sub>	0.0007 <sub>0.0003</sub>	0.777 <sub>0.001</sub>	<u>0.017<sub>0.001</sub></u>	1.01 <sub>0.00</sub>
	L-CP	<b>0.0018<sub>0.0027</sub></b>	<b>−0.0003<sub>0.0000</sub></b>	0.782 <sub>0.000</sub>	<b>0.015<sub>0.001</sub></b>	0.93 <sub>0.00</sub>
	STDQR	0.0171 <sub>0.0012</sub>	0.0003 <sub>0.0001</sub>	0.764 <sub>0.000</sub>	0.018 <sub>0.000</sub>	0.85 <sub>0.00</sub>
	CopulaCPTS	0.0128 <sub>0.0001</sub>	0.0002 <sub>0.0000</sub>	0.792 <sub>0.000</sub>	0.021 <sub>0.000</sub>	0.89 <sub>0.00</sub>
bank8FM	MSE	0.0976 <sub>0.0082</sub>	0.0249 <sub>0.0026</sub>	0.535 <sub>0.029</sub>	0.043 <sub>0.010</sub>	0.83 <sub>0.04</sub>
	MVCS	0.0605 <sub>0.0089</sub>	0.0091 <sub>0.0025</sub>	0.639 <sub>0.038</sub>	0.021 <sub>0.008</sub>	<b>0.71<sub>0.01</sub></b>
	HR	0.0244 <sub>0.0000</sub>	0.0007 <sub>0.0000</sub>	0.744 <sub>0.000</sub>	0.029 <sub>0.000</sub>	0.88 <sub>0.00</sub>
	DR-CP	0.0588 <sub>0.0007</sub>	0.0045 <sub>0.0001</sub>	0.744 <sub>0.000</sub>	0.034 <sub>0.000</sub>	<u>0.71<sub>0.00</sub></u>
	C-HDR	0.0330 <sub>0.0001</sub>	0.0007 <sub>0.0001</sub>	<b>0.801<sub>0.003</sub></b>	0.032 <sub>0.000</sub>	0.97 <sub>0.00</sub>
	PCP	0.0357 <sub>0.0003</sub>	0.0023 <sub>0.0001</sub>	0.718 <sub>0.001</sub>	0.032 <sub>0.001</sub>	0.83 <sub>0.00</sub>
	HD-PCP	0.0322 <sub>0.0021</sub>	0.0017 <sub>0.0003</sub>	0.768 <sub>0.001</sub>	0.025 <sub>0.001</sub>	0.80 <sub>0.00</sub>
	C-PCP	0.0215 <sub>0.0035</sub>	0.0003 <sub>0.0000</sub>	<u>0.780<sub>0.000</sub></u>	0.021 <sub>0.000</sub>	1.07 <sub>0.00</sub>
	CP2-PCP	<u>0.0136<sub>0.0021</sub></u>	<b>−0.0001<sub>0.0000</sub></b>	0.755 <sub>0.003</sub>	0.020 <sub>0.000</sub>	1.01 <sub>0.00</sub>
	L-CP	0.0231 <sub>0.0003</sub>	<u>0.0002<sub>0.0001</sub></u>	0.745 <sub>0.004</sub>	0.033 <sub>0.000</sub>	0.94 <sub>0.00</sub>
	STDQR	<b>0.0016<sub>0.0018</sub></b>	0.0003 <sub>0.0002</sub>	0.740 <sub>0.001</sub>	0.019 <sub>0.000</sub>	0.82 <sub>0.00</sub>
	CopulaCPTS	0.0237 <sub>0.0000</sub>	0.0011 <sub>0.0000</sub>	0.761 <sub>0.000</sub>	<b>0.015<sub>0.000</sub></b>	0.87 <sub>0.00</sub>

Dataset	Method	$L_1$ -ERT	$L_2$ -ERT	WSC	WCovGap	Size
cpu-act	MSE	0.0752 <sub>0.0108</sub>	0.0142 <sub>0.0034</sub>	0.612 <sub>0.034</sub>	0.049 <sub>0.005</sub>	1.08 <sub>0.02</sub>
	MVCS	0.0764 <sub>0.0036</sub>	0.0238 <sub>0.0022</sub>	0.671 <sub>0.012</sub>	0.027 <sub>0.003</sub>	1.05 <sub>0.04</sub>
	HR	<b>0.0054</b> <sub>0.0015</sub>	<b>-0.0006</b> <sub>0.0000</sub>	0.754 <sub>0.005</sub>	<u>0.018</u> <sub>0.002</sub>	1.01 <sub>0.04</sub>
	DR-CP	0.0276 <sub>0.0072</sub>	0.0025 <sub>0.0013</sub>	0.770 <sub>0.005</sub>	0.024 <sub>0.001</sub>	<b>0.88</b> <sub>0.00</sub>
	C-HDR	0.0353 <sub>0.0040</sub>	0.0019 <sub>0.0005</sub>	0.735 <sub>0.011</sub>	0.039 <sub>0.005</sub>	1.01 <sub>0.01</sub>
	PCP	0.0298 <sub>0.0040</sub>	0.0003 <sub>0.0005</sub>	<b>0.777</b> <sub>0.012</sub>	0.031 <sub>0.001</sub>	1.00 <sub>0.01</sub>
	HD-PCP	0.0215 <sub>0.0005</sub>	0.0002 <sub>0.0000</sub>	<u>0.771</u> <sub>0.004</sub>	0.024 <sub>0.001</sub>	<u>0.91</u> <sub>0.00</sub>
	C-PCP	0.0275 <sub>0.0172</sub>	0.0007 <sub>0.0008</sub>	0.766 <sub>0.005</sub>	0.033 <sub>0.007</sub>	1.19 <sub>0.04</sub>
	CP2-PCP	<u>0.0166</u> <sub>0.0034</sub>	<u>-0.0001</u> <sub>0.0001</sub>	0.739 <sub>0.007</sub>	<b>0.015</b> <sub>0.001</sub>	1.02 <sub>0.00</sub>
	L-CP	0.0400 <sub>0.0028</sub>	0.0015 <sub>0.0003</sub>	0.741 <sub>0.003</sub>	0.024 <sub>0.003</sub>	1.10 <sub>0.04</sub>
	STDQR	0.0272 <sub>0.0047</sub>	0.0007 <sub>0.0003</sub>	0.768 <sub>0.008</sub>	0.022 <sub>0.002</sub>	0.93 <sub>0.00</sub>
	CopulaCPTS	0.0190 <sub>0.0005</sub>	0.0007 <sub>0.0000</sub>	0.743 <sub>0.009</sub>	0.018 <sub>0.000</sub>	1.01 <sub>0.03</sub>
house-8L	MSE	0.0543 <sub>0.0051</sub>	0.0047 <sub>0.0007</sub>	0.723 <sub>0.013</sub>	0.039 <sub>0.004</sub>	1.53 <sub>0.01</sub>
	MVCS	0.0406 <sub>0.0076</sub>	0.0029 <sub>0.0007</sub>	0.763 <sub>0.012</sub>	0.023 <sub>0.004</sub>	1.47 <sub>0.01</sub>
	HR	<u>0.0138</u> <sub>0.0007</sub>	<u>0.0001</u> <sub>0.0001</sub>	0.768 <sub>0.011</sub>	0.020 <sub>0.003</sub>	1.51 <sub>0.01</sub>
	DR-CP	0.0351 <sub>0.0031</sub>	0.0018 <sub>0.0004</sub>	0.770 <sub>0.016</sub>	0.025 <sub>0.006</sub>	<b>1.35</b> <sub>0.03</sub>
	C-HDR	<b>0.0110</b> <sub>0.0008</sub>	<b>0.0001</b> <sub>0.0000</sub>	<b>0.807</b> <sub>0.005</sub>	<b>0.013</b> <sub>0.004</sub>	1.47 <sub>0.04</sub>
	PCP	0.0222 <sub>0.0079</sub>	0.0008 <sub>0.0003</sub>	0.786 <sub>0.007</sub>	0.024 <sub>0.003</sub>	1.65 <sub>0.01</sub>
	HD-PCP	0.0231 <sub>0.0004</sub>	0.0006 <sub>0.0002</sub>	0.797 <sub>0.017</sub>	0.019 <sub>0.004</sub>	<u>1.41</u> <sub>0.03</sub>
	C-PCP	0.0158 <sub>0.0015</sub>	0.0003 <sub>0.0001</sub>	<u>0.806</u> <sub>0.017</sub>	<u>0.015</u> <sub>0.000</sub>	1.72 <sub>0.01</sub>
	CP2-PCP	0.0160 <sub>0.0018</sub>	0.0004 <sub>0.0001</sub>	0.799 <sub>0.015</sub>	0.016 <sub>0.001</sub>	1.74 <sub>0.03</sub>
	L-CP	0.0214 <sub>0.0109</sub>	0.0032 <sub>0.0026</sub>	0.752 <sub>0.024</sub>	0.023 <sub>0.010</sub>	1.70 <sub>0.03</sub>
	STDQR	0.0219 <sub>0.0021</sub>	0.0003 <sub>0.0000</sub>	0.782 <sub>0.013</sub>	0.019 <sub>0.003</sub>	1.45 <sub>0.00</sub>
	CopulaCPTS	0.0269 <sub>0.0042</sub>	0.0007 <sub>0.0003</sub>	0.770 <sub>0.007</sub>	0.022 <sub>0.003</sub>	1.51 <sub>0.00</sub>
sulfur	MSE	0.0201 <sub>0.0063</sub>	0.0001 <sub>0.0003</sub>	0.753 <sub>0.017</sub>	0.021 <sub>0.005</sub>	1.45 <sub>0.04</sub>
	MVCS	0.0116 <sub>0.0099</sub>	<u>-0.0002</u> <sub>0.0007</sub>	0.744 <sub>0.020</sub>	0.025 <sub>0.003</sub>	<b>1.41</b> <sub>0.04</sub>
	HR	0.0061 <sub>0.0010</sub>	<u>-0.0000</u> <sub>0.0001</sub>	0.788 <sub>0.002</sub>	0.016 <sub>0.000</sub>	1.54 <sub>0.00</sub>
	DR-CP	<b>0.0001</b> <sub>0.0024</sub>	<u>-0.0005</u> <sub>0.0001</sub>	0.762 <sub>0.000</sub>	0.018 <sub>0.000</sub>	1.52 <sub>0.00</sub>
	C-HDR	0.0186 <sub>0.0067</sub>	<b>-0.0009</b> <sub>0.0007</sub>	0.789 <sub>0.009</sub>	0.026 <sub>0.005</sub>	1.61 <sub>0.03</sub>
	PCP	0.0159 <sub>0.0022</sub>	0.0004 <sub>0.0000</sub>	<b>0.795</b> <sub>0.002</sub>	0.028 <sub>0.001</sub>	1.75 <sub>0.01</sub>
	HD-PCP	0.0126 <sub>0.0011</sub>	<u>-0.0005</u> <sub>0.0001</sub>	<u>0.793</u> <sub>0.004</sub>	0.028 <sub>0.001</sub>	1.56 <sub>0.00</sub>
	C-PCP	0.0037 <sub>0.0077</sub>	<u>-0.0006</u> <sub>0.0001</sub>	0.771 <sub>0.005</sub>	0.017 <sub>0.003</sub>	1.80 <sub>0.02</sub>
	CP2-PCP	0.0027 <sub>0.0030</sub>	<u>-0.0008</u> <sub>0.0002</sub>	0.791 <sub>0.002</sub>	0.021 <sub>0.001</sub>	1.71 <sub>0.00</sub>
	L-CP	0.0230 <sub>0.0026</sub>	<u>0.0007</u> <sub>0.0001</sub>	0.789 <sub>0.009</sub>	0.032 <sub>0.002</sub>	1.62 <sub>0.01</sub>
	STDQR	0.0062 <sub>0.0003</sub>	<u>-0.0005</u> <sub>0.0000</sub>	0.784 <sub>0.004</sub>	<b>0.013</b> <sub>0.000</sub>	1.52 <sub>0.00</sub>
	CopulaCPTS	<u>0.0007</u> <sub>0.0025</sub>	<u>-0.0004</u> <sub>0.0001</sub>	0.753 <sub>0.003</sub>	<u>0.016</u> <sub>0.001</sub>	1.51 <sub>0.00</sub>

**Multivariate regression** The metrics EOC and SSC are used to compute the CovGap.

Dataset	Method	$L_1$ -ERT	$L_2$ -ERT	WSC	WCovGap	Size
CASP	MSE	0.0407 <sub>0.0040</sub>	0.0023 <sub>0.0004</sub>	0.818 <sub>0.007</sub>	0.025 <sub>0.003</sub>	1.26 <sub>0.01</sub>
	MVCS	0.0304 <sub>0.0039</sub>	0.0016 <sub>0.0005</sub>	0.834 <sub>0.006</sub>	0.016 <sub>0.002</sub>	<b>1.23</b> <sub>0.01</sub>
	HR	0.0252 <sub>0.0000</sub>	0.0008 <sub>0.0000</sub>	0.824 <sub>0.000</sub>	0.020 <sub>0.000</sub>	1.32 <sub>0.00</sub>
	DR-CP	0.0177 <sub>0.0000</sub>	0.0005 <sub>0.0000</sub>	0.836 <sub>0.000</sub>	0.017 <sub>0.000</sub>	1.25 <sub>0.00</sub>
	C-HDR	0.0235 <sub>0.0000</sub>	0.0008 <sub>0.0000</sub>	<b>0.862</b> <sub>0.000</sub>	0.021 <sub>0.000</sub>	1.35 <sub>0.00</sub>
	PCP	0.0147 <sub>0.0000</sub>	0.0003 <sub>0.0000</sub>	0.827 <sub>0.000</sub>	0.022 <sub>0.000</sub>	1.41 <sub>0.00</sub>
	HD-PCP	0.0238 <sub>0.0000</sub>	0.0006 <sub>0.0000</sub>	0.817 <sub>0.000</sub>	0.018 <sub>0.000</sub>	1.30 <sub>0.00</sub>
	C-PCP	0.0232 <sub>0.0000</sub>	0.0004 <sub>0.0000</sub>	0.845 <sub>0.000</sub>	<b>0.009</b> <sub>0.000</sub>	1.44 <sub>0.00</sub>
	CP2-PCP	0.0154 <sub>0.0000</sub>	0.0003 <sub>0.0000</sub>	0.845 <sub>0.000</sub>	0.016 <sub>0.000</sub>	1.43 <sub>0.00</sub>
	L-CP	<b>0.0117</b> <sub>0.0000</sub>	<u>0.0002</u> <sub>0.0000</sub>	<u>0.846</u> <sub>0.000</sub>	<u>0.009</u> <sub>0.000</sub>	1.42 <sub>0.00</sub>
	STDQR	0.0158 <sub>0.0000</sub>	<u>0.0004</u> <sub>0.0000</sub>	0.819 <sub>0.000</sub>	0.016 <sub>0.000</sub>	1.32 <sub>0.00</sub>
	CopulaCPTS	<u>0.0140</u> <sub>0.0000</sub>	<b>0.0002</b> <sub>0.0000</sub>	0.831 <sub>0.000</sub>	0.013 <sub>0.000</sub>	1.34 <sub>0.00</sub>
bias	MSE	0.0268 <sub>0.0174</sub>	0.0011 <sub>0.0010</sub>	<u>0.722</u> <sub>0.026</sub>	0.017 <sub>0.005</sub>	<b>1.03</b> <sub>0.01</sub>
	MVCS	0.0219 <sub>0.0096</sub>	0.0005 <sub>0.0009</sub>	<b>0.726</b> <sub>0.021</sub>	<b>0.014</b> <sub>0.004</sub>	<u>1.04</u> <sub>0.02</sub>
	HR	0.0147 <sub>0.0000</sub>	<u>-0.0003</u> <sub>0.0000</sub>	0.709 <sub>0.000</sub>	0.029 <sub>0.000</sub>	1.09 <sub>0.00</sub>
	DR-CP	0.0338 <sub>0.0000</sub>	0.0027 <sub>0.0000</sub>	0.679 <sub>0.000</sub>	0.031 <sub>0.000</sub>	1.20 <sub>0.00</sub>
	C-HDR	0.0387 <sub>0.0000</sub>	0.0018 <sub>0.0000</sub>	0.696 <sub>0.000</sub>	0.034 <sub>0.000</sub>	1.36 <sub>0.00</sub>
	PCP	0.0181 <sub>0.0000</sub>	<u>-0.0003</u> <sub>0.0000</sub>	0.705 <sub>0.000</sub>	0.024 <sub>0.000</sub>	1.31 <sub>0.00</sub>
	HD-PCP	0.0183 <sub>0.0000</sub>	0.0008 <sub>0.0000</sub>	0.692 <sub>0.000</sub>	0.029 <sub>0.000</sub>	1.19 <sub>0.00</sub>
	C-PCP	<b>0.0028</b> <sub>0.0000</sub>	<b>-0.0006</b> <sub>0.0000</sub>	0.705 <sub>0.000</sub>	0.027 <sub>0.000</sub>	1.41 <sub>0.00</sub>
	CP2-PCP	0.0199 <sub>0.0000</sub>	0.0029 <sub>0.0000</sub>	0.667 <sub>0.000</sub>	0.039 <sub>0.000</sub>	1.30 <sub>0.00</sub>
	L-CP	0.0485 <sub>0.0000</sub>	0.0033 <sub>0.0000</sub>	0.646 <sub>0.000</sub>	0.042 <sub>0.000</sub>	1.24 <sub>0.00</sub>
	STDQR	<u>0.0139</u> <sub>0.0000</sub>	<u>-0.0000</u> <sub>0.0000</sub>	0.692 <sub>0.000</sub>	0.022 <sub>0.000</sub>	1.19 <sub>0.00</sub>
	CopulaCPTS	0.0446 <sub>0.0000</sub>	0.0023 <sub>0.0000</sub>	0.675 <sub>0.000</sub>	0.044 <sub>0.000</sub>	1.21 <sub>0.00</sub>
house	MSE	0.0599 <sub>0.0070</sub>	0.0064 <sub>0.0017</sub>	0.682 <sub>0.021</sub>	0.044 <sub>0.006</sub>	1.09 <sub>0.02</sub>
	MVCS	0.0605 <sub>0.0049</sub>	0.0071 <sub>0.0009</sub>	0.713 <sub>0.018</sub>	0.038 <sub>0.005</sub>	<b>1.02</b> <sub>0.03</sub>
	HR	0.0339 <sub>0.0038</sub>	0.0026 <sub>0.0006</sub>	0.740 <sub>0.012</sub>	0.038 <sub>0.001</sub>	1.17 <sub>0.02</sub>
	DR-CP	0.0450 <sub>0.0001</sub>	0.0037 <sub>0.0002</sub>	0.754 <sub>0.002</sub>	0.026 <sub>0.000</sub>	<u>1.07</u> <sub>0.00</sub>
	C-HDR	<u>0.0167</u> <sub>0.0054</sub>	0.0008 <sub>0.0001</sub>	0.758 <sub>0.000</sub>	0.021 <sub>0.000</sub>	1.11 <sub>0.00</sub>
	PCP	0.0513 <sub>0.0024</sub>	0.0049 <sub>0.0002</sub>	0.740 <sub>0.000</sub>	0.021 <sub>0.001</sub>	1.21 <sub>0.00</sub>
	HD-PCP	0.0361 <sub>0.0037</sub>	0.0023 <sub>0.0001</sub>	0.746 <sub>0.007</sub>	0.026 <sub>0.001</sub>	1.12 <sub>0.00</sub>
	C-PCP	0.0181 <sub>0.0030</sub>	<b>0.0005</b> <sub>0.0000</sub>	0.770 <sub>0.003</sub>	0.020 <sub>0.000</sub>	1.28 <sub>0.00</sub>
	CP2-PCP	<b>0.0166</b> <sub>0.0042</sub>	<u>0.0005</u> <sub>0.0002</sub>	<b>0.790</b> <sub>0.004</sub>	<u>0.018</u> <sub>0.002</sub>	1.27 <sub>0.00</sub>
	L-CP	0.0286 <sub>0.0006</sub>	0.0018 <sub>0.0001</sub>	<u>0.774</u> <sub>0.000</sub>	0.020 <sub>0.000</sub>	1.22 <sub>0.00</sub>
	STDQR	0.0382 <sub>0.0006</sub>	0.0027 <sub>0.0001</sub>	0.751 <sub>0.007</sub>	0.028 <sub>0.000</sub>	1.14 <sub>0.00</sub>
	CopulaCPTS	0.0212 <sub>0.0013</sub>	0.0008 <sub>0.0001</sub>	0.766 <sub>0.002</sub>	<b>0.013</b> <sub>0.000</sub>	1.22 <sub>0.00</sub>

Dataset	Method	$L_1$ -ERT	$L_2$ -ERT	WSC	WCovGap	Size
rf1	MSE	0.0647 <sub>0.0063</sub>	0.0068 <sub>0.0020</sub>	0.649 <sub>0.030</sub>	0.060 <sub>0.006</sub>	0.95 <sub>0.00</sub>
	MVCS	0.1011 <sub>0.0074</sub>	0.0226 <sub>0.0034</sub>	0.588 <sub>0.032</sub>	0.071 <sub>0.007</sub>	<b>0.40<sub>0.01</sub></b>
	HR	<u>0.0372<sub>0.0134</sub></u>	<u>0.0029<sub>0.0025</sub></u>	0.717 <sub>0.027</sub>	0.036 <sub>0.000</sub>	0.57 <sub>0.02</sub>
	DR-CP	0.0860 <sub>0.0016</sub>	0.0132 <sub>0.0002</sub>	0.689 <sub>0.010</sub>	0.053 <sub>0.009</sub>	0.46 <sub>0.02</sub>
	C-HDR	0.0484 <sub>0.0059</sub>	0.0032 <sub>0.0001</sub>	<b>0.765<sub>0.021</sub></b>	0.030 <sub>0.010</sub>	0.46 <sub>0.03</sub>
	PCP	0.0958 <sub>0.0005</sub>	0.0101 <sub>0.0006</sub>	0.656 <sub>0.009</sub>	0.075 <sub>0.001</sub>	0.48 <sub>0.01</sub>
	HD-PCP	0.0972 <sub>0.0066</sub>	0.0194 <sub>0.0030</sub>	0.612 <sub>0.004</sub>	0.070 <sub>0.005</sub>	0.47 <sub>0.01</sub>
	C-PCP	<b>0.0324<sub>0.0088</sub></b>	<b>0.0019<sub>0.0017</sub></b>	<u>0.744<sub>0.009</sub></u>	<b>0.028<sub>0.004</sub></b>	0.48 <sub>0.01</sub>
	CP2-PCP	0.0448 <sub>0.0121</sub>	0.0042 <sub>0.0030</sub>	0.717 <sub>0.000</sub>	<u>0.029<sub>0.013</sub></u>	0.46 <sub>0.02</sub>
	L-CP	0.0550 <sub>0.0004</sub>	0.0059 <sub>0.0014</sub>	0.702 <sub>0.005</sub>	0.030 <sub>0.013</sub>	0.46 <sub>0.02</sub>
rf2	STDQR	0.0934 <sub>0.0113</sub>	0.0152 <sub>0.0049</sub>	0.637 <sub>0.021</sub>	0.073 <sub>0.004</sub>	0.48 <sub>0.01</sub>
	CopulaCPTS	0.0759 <sub>0.0005</sub>	0.0087 <sub>0.0023</sub>	0.733 <sub>0.075</sub>	0.038 <sub>0.006</sub>	0.48 <sub>0.02</sub>
	MSE	0.0498 <sub>0.0096</sub>	0.0033 <sub>0.0021</sub>	0.737 <sub>0.016</sub>	0.024 <sub>0.003</sub>	0.95 <sub>0.00</sub>
	MVCS	0.1100 <sub>0.0097</sub>	0.0329 <sub>0.0058</sub>	0.573 <sub>0.042</sub>	0.076 <sub>0.009</sub>	0.53 <sub>0.02</sub>
	HR	0.0557 <sub>0.0019</sub>	0.0110 <sub>0.0001</sub>	0.714 <sub>0.014</sub>	0.020 <sub>0.005</sub>	1.07 <sub>0.01</sub>
	DR-CP	0.0669 <sub>0.0015</sub>	0.0128 <sub>0.0001</sub>	0.727 <sub>0.020</sub>	0.022 <sub>0.001</sub>	<b>0.39<sub>0.00</sub></b>
	C-HDR	0.0524 <sub>0.0099</sub>	0.0049 <sub>0.0021</sub>	<b>0.757<sub>0.004</sub></b>	0.021 <sub>0.001</sub>	0.40 <sub>0.00</sub>
	PCP	0.0825 <sub>0.0007</sub>	0.0145 <sub>0.0027</sub>	0.701 <sub>0.005</sub>	0.025 <sub>0.007</sub>	0.42 <sub>0.01</sub>
	HD-PCP	0.0722 <sub>0.0035</sub>	0.0115 <sub>0.0004</sub>	0.739 <sub>0.011</sub>	0.023 <sub>0.000</sub>	0.42 <sub>0.00</sub>
	C-PCP	<u>0.0366<sub>0.0053</sub></u>	<b>0.0011<sub>0.0000</sub></b>	0.749 <sub>0.021</sub>	0.016 <sub>0.004</sub>	0.42 <sub>0.00</sub>
taxi	CP2-PCP	<b>0.0291<sub>0.0081</sub></b>	<u>0.0013<sub>0.0004</sub></u>	<u>0.750<sub>0.022</sub></u>	<u>0.015<sub>0.004</sub></u>	0.42 <sub>0.00</sub>
	L-CP	0.0580 <sub>0.0186</sub>	<u>0.0077<sub>0.0030</sub></u>	0.714 <sub>0.015</sub>	<b>0.013<sub>0.001</sub></b>	0.40 <sub>0.00</sub>
	STDQR	0.0702 <sub>0.0079</sub>	0.0095 <sub>0.0016</sub>	0.743 <sub>0.004</sub>	0.017 <sub>0.004</sub>	0.42 <sub>0.00</sub>
	CopulaCPTS	0.0643 <sub>0.0024</sub>	0.0122 <sub>0.0026</sub>	0.707 <sub>0.087</sub>	0.031 <sub>0.002</sub>	1.43 <sub>1.02</sub>
	MSE	0.0234 <sub>0.0031</sub>	0.0011 <sub>0.0001</sub>	0.815 <sub>0.003</sub>	0.018 <sub>0.001</sub>	<b>1.88<sub>0.00</sub></b>
	MVCS	0.0195 <sub>0.0027</sub>	<u>0.0007<sub>0.0002</sub></u>	0.816 <sub>0.006</sub>	0.020 <sub>0.001</sub>	3.12 <sub>0.01</sub>
	HR	0.0044 <sub>0.0025</sub>	0.0001 <sub>0.0000</sub>	0.844 <sub>0.002</sub>	<b>0.007<sub>0.002</sub></b>	3.48 <sub>0.00</sub>
	DR-CP	0.0197 <sub>0.0015</sub>	0.0005 <sub>0.0002</sub>	0.830 <sub>0.004</sub>	0.014 <sub>0.001</sub>	<u>2.70<sub>0.01</sub></u>
	C-HDR	0.0128 <sub>0.0002</sub>	0.0002 <sub>0.0000</sub>	0.840 <sub>0.004</sub>	0.011 <sub>0.001</sub>	2.74 <sub>0.00</sub>
	PCP	0.0152 <sub>0.0023</sub>	0.0007 <sub>0.0001</sub>	0.808 <sub>0.004</sub>	0.020 <sub>0.003</sub>	3.20 <sub>0.01</sub>
adult	HD-PCP	0.0106 <sub>0.0011</sub>	0.0004 <sub>0.0002</sub>	0.833 <sub>0.005</sub>	0.015 <sub>0.002</sub>	2.99 <sub>0.02</sub>
	C-PCP	0.0112 <sub>0.0068</sub>	0.0001 <sub>0.0001</sub>	0.846 <sub>0.001</sub>	<u>0.009<sub>0.001</sub></u>	3.26 <sub>0.01</sub>
	CP2-PCP	<b>-0.0026<sub>0.0047</sub></b>	<b>-0.0000<sub>0.0000</sub></b>	<b>0.849<sub>0.007</sub></b>	0.012 <sub>0.002</sub>	3.24 <sub>0.02</sub>
	L-CP	0.0088 <sub>0.0006</sub>	0.0001 <sub>0.0000</sub>	0.832 <sub>0.000</sub>	0.009 <sub>0.000</sub>	3.04 <sub>0.00</sub>
	STDQR	0.0053 <sub>0.0004</sub>	0.0002 <sub>0.0001</sub>	0.839 <sub>0.002</sub>	0.010 <sub>0.000</sub>	3.04 <sub>0.01</sub>
	CopulaCPTS	0.0030 <sub>0.0026</sub>	<u>0.0000<sub>0.0000</sub></u>	0.847 <sub>0.000</sub>	0.010 <sub>0.001</sub>	3.38 <sub>0.03</sub>