CONFORMAL PREDICTION AND CONDITIONAL COVERAGE FOR TABULAR FOUNDATION MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Tabular foundation models (TFMs) provide predictive distributions for regression, but their prediction regions can exhibit undercoverage or overcoverage even when point predictions are accurate. We introduce C-USIM (Conditionally-Uniformized Score Integration Method), a lightweight application of highest predictive density split conformal prediction that accommodates multimodal predictions. Given calibration and test outputs, it requires no additional training or model inference. It provides finite-sample marginal validity under our assumptions. We bound conditional–marginal coverage gaps using distribution-estimation error and score discreteness, and examine coverage heterogeneity through percentile rank–score plots. Experiments with TabPFN and TabICL show improved marginal coverage accuracy and lower average conditional and group coverage errors. Under a fixed data budget, allocating more observations to calibration can reduce marginal coverage error despite less accurate point predictions.

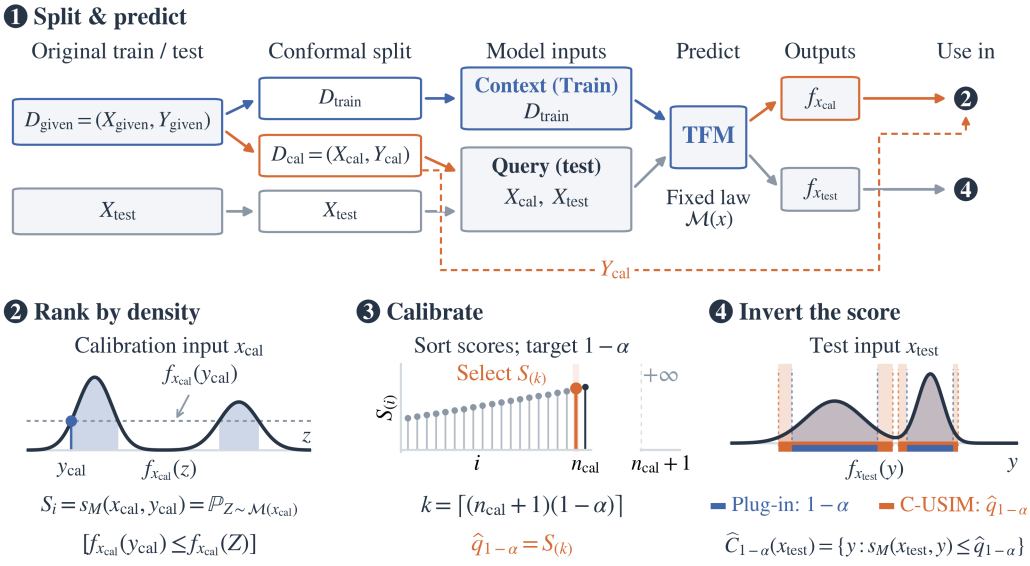

Figure 1: C-USIM overview with n_{cal} calibration scores and target coverage $1-\alpha$. Using a pretrained TFM, the procedure predicts densities, computes density-rank scores, calibrates a threshold, and constructs prediction regions without fitting an additional model. See Section 4 for details.

1 INTRODUCTION

Tabular foundation models (TFMs) built on prior-data fitted networks (PFNs) approximate posterior predictive distributions (PPD) through pretraining on synthetic data and in-context learning from a labeled context (Müller et al., 2024). Mismatch between pretraining and downstream distributions, together with approximation errors, can overestimate or underestimate these predictions despite accurate point estimates, motivating calibration and coverage assessment across inputs.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

We introduce C-USIM (Conditionally-Uniformized Score Integration Method), a split conformal procedure that calibrates highest predictive density (HPD) regions from TFM outputs. HPD regions can comprise disjoint intervals for multimodal predictions. Uncalibrated plug-in HPD can under- or overcover when the predictive distribution is inaccurate. Following HPD-split (Izbicki et al., 2022), C-USIM sets its density-rank threshold to an empirical quantile of held-out scores. Calibration requires no additional training, model inference, or separate density or score-correction model.

Under exchangeability, calibration guarantees finite-sample marginal coverage of at least the target but can leave systematic undercoverage in some covariate regions and overcoverage in others (Vovk et al., 2005). We analyze conditional–marginal coverage gaps through distribution-estimation error and score discreteness, using percentile rank–score plots to show coverage improvements after calibration and remaining variation across inputs.

Under a fixed label budget, we compare reserving labels for calibration with using all labels as context for uncalibrated plug-in HPD. TabPFN and TabICL experiments show improved marginal coverage accuracy, lower mean absolute conditional coverage error across synthetic mechanisms, and lower aggregate group coverage errors in most real-world comparisons. We further examine the training–calibration ratio within C-USIM and find that allocating more labels to calibration can improve coverage accuracy despite less accurate point predictions.

We make three contributions. **First,** we formulate C-USIM by applying HPD-split to TFM bin-probability and quantile outputs, calibrating potentially disjoint prediction regions without additional model fitting or inference. **Second,** under the stated assumptions, we bound conditional–marginal coverage gaps using bin-probability estimation error and score discreteness, and relate a shared threshold to coverage at individual inputs through our percentile rank–score representation. **Third,** our experiments under a fixed label budget provide evidence of coverage improvements over plug-in HPD and guidance on context–calibration allocation: more calibration observations can improve marginal coverage accuracy despite less accurate point predictions.

2 PRELIMINARIES

2.1 PRIOR-DATA FITTED NETWORK

Given a training dataset $D_{\text{train}} = \{(x_1, y_1), \cdots, (x_{n_{\text{train}}}, y_{n_{\text{train}}})\}$, we estimate the conditional distribution of the response Y at $X = x_{\text{test}}$. In the Bayesian framework for supervised learning, let ϕ denote a data-generating mechanism in a space Φ, sampled from the prior $p(\phi)$, with datasets D generated independently conditional on ϕ. The PPD $p(\cdot \mid x_{\text{test}}, D_{\text{train}})$ marginalizes over the posterior of ϕ:

$$p(y \mid x, D) \propto \int_{\Phi} p(y \mid x, \phi) p(D \mid \phi) p(\phi) d\phi$$

Prior-data Fitted Networks (PFNs) use Transformers pretrained on synthetic datasets to approximate this Bayesian inference, predicting the PPD from a training context. In univariate regression, where $y \in \mathbb{R}$, TabPFN learns probability masses over response bins using cross-entropy loss (Grinsztajn et al., 2026), whereas TabICL estimates conditional quantiles using pinball loss (Qu et al., 2026).

2.2 PLUG-IN HPD REGIONS

Let f_x denote an estimated conditional density of Y given $X = x$. For $0 < \alpha < 1$, the plug-in HPD region is defined by the highest-density-region construction (Hyndman, 1996):

$$\widehat{C}^{\text{plug}}_{1-\alpha}(x) := \{y : f_x(y) \geq \lambda_\alpha(x)\},$$
$$\lambda_\alpha(x) := \sup\left\{t > 0 : \int_{\{z : f_x(z) \geq t\}} f_x(z)\, dz \geq 1 - \alpha\right\}.$$

The cutoff $\lambda_\alpha(x)$ retains at least $1 - \alpha$ predictive mass at the largest possible density threshold. For piecewise-constant densities, whole positive-density levels are included in decreasing order until

their cumulative mass first reaches or exceeds the target, retaining all boundary ties. These potentially disconnected regions require no calibration, and their predictive mass does not guarantee coverage under the true data distribution.

2.3 Split Conformal Prediction

Split conformal prediction (CP) guarantees finite-sample marginal coverage under exchangeability (Vovk et al., 2005). After fitting a model on training data, a held-out calibration set $\{(X_i, Y_i)\}_{i=1}^{n_{\text{cal}}}$ and a nonconformity rule $s(x, y)$ yield scores $\{S_i = s(X_i, Y_i)\}_{i=1}^{n_{\text{cal}}}$. For target coverage $1 - \alpha$, define $\widehat{C}_{1-\alpha}(x) := \{y : s(x, y) \leq \widehat{q}_{1-\alpha}\}$ where $\widehat{q}_{1-\alpha}$ is the $\lceil (n_{\text{cal}} + 1)(1 - \alpha) \rceil$th order statistic of $\{S_i\}_{i=1}^{n_{\text{cal}}} \cup \{\infty\}$. However, nontrivial distribution-free guarantees of exact conditional coverage are generally impossible in finite samples without additional assumptions (Barber et al., 2021).

For a real-valued random variable W with CDF $F_W(w)$, the probability integral transform (PIT) $U := F_W(W)$ is uniform when F_W is continuous: $U \sim \text{Unif}(0, 1)$. A score is *pivotal* if its conditional law is independent of the covariate. For $S = s(X, Y)$ with conditional CDF $F_{S|X=x}$, if $F_{S|X=x}$ is continuous for every x, the oracle conditional PIT correction $\tilde{s}(x, y) := F_{S|X=x}(s(x, y))$ satisfies $\tilde{s}(X, Y) \mid X = x \sim \text{Unif}(0, 1)$. Since this conditional distribution is the same across covariates, the corrected score is pivotal. The region $\{y : \tilde{s}(x, y) \leq 1 - \alpha\}$ therefore has conditional coverage $1 - \alpha$ for every x (Laplante, 2026).

C-USIM computes the model-based score CDF from the TFM predictive distribution without fitting a separate conditional score model. Piecewise-constant predictive densities can also induce discrete scores whose deterministic PIT need not be uniform even under the predictive model. We therefore analyze distribution-estimation error and score discreteness.

3 Related work

Conformal prediction provides distribution-free marginal coverage guarantees (Gammerman et al., 1998; Saunders et al., 1999; Vovk et al., 2005), with extensive work on regression (Shafer & Vovk, 2008; Lei & Wasserman, 2014). HPD-split uses density-based scores to construct adaptive prediction regions for heteroscedastic, skewed, or multimodal conditional distributions (Izbicki et al., 2022). Its HPD-based score is also studied in the C-HDR formulation alongside a broader family of CDF-based conformity scores (Dheur et al., 2025).

Conformalized quantile regression (CQR) calibrates residuals around estimated lower and upper conditional quantiles to form adaptive prediction intervals (Romano et al., 2019). Conformalized histogram regression (CHR) calibrates a nested family of shortest contiguous intervals from conditional histograms using region probability mass as the conformity score; its interval construction can affect length and computational cost (Sesia & Romano, 2021). When applied to the same TFM predictive distribution, CQR and CHR return contiguous intervals, which include the intervening low-density region whenever they cover separated modes. C-USIM instead permits disjoint high-density regions, allowing such gaps to be excluded and adapting prediction-set geometry to multimodal predictive densities. This flexibility allows C-USIM to handle broader families of distributions, a capability that is particularly important for TFMs.

JAPAN thresholds densities estimated by normalizing flows, while DSPS combines conditional normalizing flows with density-based ranking and conformal calibration (English & Lippert, 2026; Luo & Zhou, 2025). PIT-CP-MDN and PIT-CP-CNF fit additional models for conditional score distributions using mixture density networks and conditional normalizing flows, respectively, to seek approximate conditional coverage for different base scores (Laplante, 2026).

Finally, for tabular models, CP has been applied to various settings using absolute-error scores for regression and adaptive prediction-set scores for classification (van Leeuwen, 2025; Costa et al., 2026; Kenfack et al., 2026). We apply HPD-split and its PIT interpretation to pretrained TFM outputs directly, without fitting an additional density or score-correction model. Our contribution combines this lightweight construction with conditional-coverage analysis, percentile rank–score diagnostics, and fixed-budget context–calibration allocation experiments.

4 C-USIM: CONDITIONALLY-UNIFORMIZED SCORE INTEGRATION METHOD

C-USIM constructs prediction regions by reconstructing densities from TFM outputs, computing density-rank scores, and calibrating a threshold on held-out observations.

4.1 SETUP AND NOTATIONS

For a random variable W, we denote its cumulative distribution function and probability density function, when it exists, by F_W and f_W, respectively.

In this paper, we consider d-dimensional covariates and a univariate response. We denote the training and calibration datasets by D_{train} and D_{cal}, consisting of n_{train} and n_{cal} covariate-response pairs in $\mathbb{R}^d \times \mathbb{R}$, respectively, drawn independently from a common distribution. We also denote a PFN model by $\mathcal{M}(\cdot, D_{\text{train}}) : \mathbb{R}^d \to \mathcal{P}(\mathbb{R})$, which uses D_{train} as context and maps each covariate x to an estimator of the conditional distribution of Y given $X = x$, assuming that this conditional distribution exists. Unless otherwise stated, we omit the second argument and write $\mathcal{M}(x)$ for $\mathcal{M}(x, D_{\text{train}})$.

Assumption 1 (Row-permutation equivariance)**.** *For fixed D_{train}, let $\mathcal{M}(\mathbf{x})$ denote the ordered predictive distributions returned jointly for the calibration and test covariates $\mathbf{x} = (x_1, \ldots, x_m)$. We assume $\mathcal{M}(\pi\mathbf{x}) = \pi\mathcal{M}(\mathbf{x})$ for every row permutation π.*

4.2 GENERAL SETTING OF C-USIM

For $x \in \mathbb{R}^d$, our goal is to construct $\widehat{C}_{1-\alpha}(x)$ from density predictions with the following properties:

- $\mathbb{P}_{(X,Y)}(Y \in \widehat{C}_{1-\alpha}(X)) \geq 1 - \alpha$, i.e., marginal coverage achieves the target.
- Conditional coverage $\mathbb{P}_Y(Y \in \widehat{C}_{1-\alpha}(x)|X = x)$ concentrates around $1 - \alpha$.
- The total length of the prediction region $\widehat{C}_{1-\alpha}(X)$ is as small as possible.

C-USIM uses the CP framework with the nonconformity score $s_{\mathcal{M}}$ obtained by applying the model-based PIT to the negated predictive density. Specifically, we take $s_0(x, y) = -f_{\mathcal{M}(x)}(y)$ as the base score and evaluate its CDF under $Z \sim \mathcal{M}(x)$ at $s_0(x, y)$. Since $s_0(x, Z) \leq s_0(x, y)$ is equivalent to $f_{\mathcal{M}(x)}(Z) \geq f_{\mathcal{M}(x)}(y)$, this gives

$$s_{\mathcal{M}}(x, y) := \mathbb{P}_{Z\sim\mathcal{M}(x)}(f_{\mathcal{M}(x)}(y) \leq f_{\mathcal{M}(x)}(Z))$$

Algorithm 1 C-USIM prediction-set construction

Require: Pretrained TFM $\mathcal{M}$, labeled data D_{given}, calibration size n_{cal}, test covariates $\mathbf{x}_{\text{test}}$, and miscoverage level $\alpha \in (0, 1)$.

Ensure: Prediction sets $\widehat{C}_{1-\alpha}(x)$ for each test covariate x.

1: Split $D_{\text{given}} = D_{\text{train}} \dot{\cup} D_{\text{cal}}$ with $|D_{\text{cal}}| = n_{\text{cal}}$.
2: Use D_{train} as labeled context for $\mathcal{M}$.
3: Query the calibration and test covariates jointly, as in Assumption 1.
4: Construct densities $f_x := f_{\mathcal{M}(x)}$ from the outputs (Appendix B.1).
5: **for** each $(X_i, Y_i) \in D_{\text{cal}}$ **do**
6: $\quad S_i \leftarrow \mathbb{P}_{Z\sim\mathcal{M}(X_i)}(f_{X_i}(Y_i) \leq f_{X_i}(Z))$.
7: **end for**
8: $k \leftarrow \lceil (n_{\text{cal}} + 1)(1 - \alpha) \rceil$.
9: $\widehat{q}_{1-\alpha} \leftarrow$ the k-th smallest value in $\{S_i\}_{i=1}^{n_{\text{cal}}} \cup \{\infty\}$.
10: **return** $\widehat{C}_{1-\alpha}(x) = \{y : s_{\mathcal{M}}(x, y) \leq \widehat{q}_{1-\alpha}\}$ for each $x \in \mathbf{x}_{\text{test}}$.

Algorithm 1 gives the complete prediction-set construction. Density-level ordering, cumulative probability masses, and an empirical calibration quantile determine the regions without additional model fitting or inference.

Models used in this work For all analyses and subsequent experiments, we use TabPFN-v3[1] and TabICL-regressor-v2[2] with the default inference settings of their respective implementations. Hereafter, we refer to these models simply as TabPFN and TabICL, respectively. For both models, we assume the permutation-equivariance condition in Assumption 1 for calibration and test rows, which is natural for their Transformer architectures with suitably masked test attention.

5 Theoretical Analysis of Conditional Coverage for C-USIM

In this section and its proofs (Appendix A), we use a stronger assumption than Assumption 1.

Assumption 2 (Test-set invariance)**.** *For fixed D_{train}, consider any two test-covariate datasets $\mathbf{x}_1 = (x_{11}, \ldots, x_{1m})$ and $\mathbf{x}_2 = (x_{21}, \ldots, x_{2n})$. For any indices p and q satisfying $x_{1p} = x_{2q}$, we assume that*

$$\mathcal{M}(x_{1p}, D_{\text{train}}; \mathbf{x}_1) = \mathcal{M}(x_{2q}, D_{\text{train}}; \mathbf{x}_2),$$

where $\mathcal{M}(x, D_{\text{train}}; \mathbf{x})$ denotes the response distribution returned for x when prediction is performed on the test dataset $\mathbf{x}$.

We assume that the calibration and test observations are i.i.d.; under Assumption 2, their scores are therefore i.i.d. evaluations of a fixed scoring rule.

5.1 Conditional Coverage Gap Bound

The conditional coverage gap of conformal prediction at x, with respect to the scoring rule s, is defined as the equation below (Laplante, 2026):

$$\Delta(x) := \sup_{\alpha \in (0,1)} \left| \mathbb{P}\left(Y \in \widehat{C}_{1-\alpha}(x) \mid X = x\right) - \mathbb{P}\left(Y \in \widehat{C}_{1-\alpha}(X)\right) \right|$$

We proved that this can be bounded in terms of the discrete L^1 error of the bin probability vectors.

Theorem 1 (Conditional coverage gap bound for the PIT-transformed score)**.** *Fix $x \in \mathbb{R}^d$, and let*

$$-\infty = \hat{y}_0^{(x)} < \hat{y}_1^{(x)} < \cdots < \hat{y}_{r+1}^{(x)} = \infty.$$

Suppose that the nonconformity score $s_{\mathcal{M}}(x, \cdot)$ is constant on each interval $[\hat{y}_{i-1}^{(x)}, \hat{y}_i^{(x)})$, with constant value $S_i^{(x)}$, and that these values are distinct across intervals. Let

$$\hat{p}_i^{(x)} := \mathbb{P}_{Z_x \sim \mathcal{M}(x)}\left(Z_x \in [\hat{y}_{i-1}^{(x)}, \hat{y}_i^{(x)})\right), \quad \tilde{p}_i^{(x)} := \mathbb{P}\left(Y \in [\hat{y}_{i-1}^{(x)}, \hat{y}_i^{(x)}) \mid X = x\right),$$

and define

$$B(x) := \frac{1}{2} \left\| \tilde{p}^{(x)} - \hat{p}^{(x)} \right\|_1 + \max_{1 \le i \le r+1} \hat{p}_i^{(x)}.$$

Here, $\tilde{p}^{(x)} = (\tilde{p}_1^{(x)}, \tilde{p}_2^{(x)}, \ldots, \tilde{p}_{r+1}^{(x)})$, $\hat{p}^{(x)} = (\hat{p}_1^{(x)}, \hat{p}_2^{(x)}, \ldots, \hat{p}_{r+1}^{(x)}) \in \mathbb{R}^{r+1}$.

Let $\tilde{s}$ be the PIT-transformed scoring rule induced by $s_{\mathcal{M}}$, and let $\Delta_{\tilde{s}}(x)$ denote the conditional coverage gap of conformal prediction constructed with $\tilde{s}$. Then,

$$\Delta_{\tilde{s}}(x) \le \mathbb{E}_X[B(X)] + B(x).$$

This result characterizes a trade-off between model complexity and the performance of CP with a PIT-corrected score. As r increases, the partition becomes finer and can reduce the discretization term $\max_{1 \le i \le r+1} \hat{p}_i^{(x)}$. However, estimating the resulting higher-dimensional probability vector may increase the L^1 error term. Conversely, using a smaller r can yield a more stable probability estimate but incurs a larger discretization error.

Since this analysis yields a global bound for $\alpha \in (0, 1)$, it does not characterize the distribution of conditional coverage over X. To extend the discussion, in the following section, we use the percentile rank–score plot to visualize and examine the distribution of conditional coverage across inputs empirically.

[1] https://github.com/priorlabs/tabpfn

[2] https://github.com/soda-inria/tabicl

5.2 Conditional Coverage Through Percentile Rank–Score Plots

Assumption 3 (Continuous score distribution)**.** *For the results in this section, we assume that the C-USIM score $s_{\mathcal{M}}(x, Y^{(x)})$ has a continuous distribution, where $Y^{(x)} \sim Y \mid X = x$ for every covariate x.*

Let $T(x, y) := s_{\mathcal{M}}(x, y), P(x, y) := F_{s_{\mathcal{M}}(X,Y)|X=x}(T(x, y))$ denote the score and its conditional percentile rank, respectively. Let Z be a conditionally independent copy of Y given X, i.e., $(Z \mid X = x) \overset{d}{=} (Y \mid X = x)$ and $Z \perp\!\!\!\perp Y \mid X$.

Under Assumption 3, $P(x, y) = \mathbb{P}(s_{\mathcal{M}}(x, Z) < s_{\mathcal{M}}(x, y) \mid X = x) = \mathbb{P}(Z \in \hat{C}_{T(x,y)}(x)|X = x)$. Hence, $P(x, y)$ can be interpreted as the true conditional coverage obtained when $T(x, y)$ is used as the score threshold. Moreover, by PIT, $P(X, Y) \mid X = x \sim \mathrm{Unif}(0, 1)$ and consequently $P(X, Y) \sim \mathrm{Unif}(0, 1)$.

For a fixed score threshold t, define

$$P_t(x) := F_{s_{\mathcal{M}}(X,Y)|X=x}(t) = \mathbb{P}\left(s_{\mathcal{M}}(X, Y) \le t \mid X = x\right) = \mathbb{P}\left(Y \in \widehat{C}_t(X) \mid X = x\right).$$

Thus, $P_t(x)$ is the conditional coverage at covariate vector x obtained by applying t as the score threshold. Thus, $P_t(X)$ describes the distribution of conditional coverage associated with $\widehat{C}_t$.

In C-USIM with target coverage $1 - \alpha$, let $\{T_i\}_{i=1}^{n_{\mathrm{cal}}+1}$ denote the calibration scores, where $T_i = T(X_i, Y_i)$ for $i = 1, \ldots, n_{\mathrm{cal}}$ and $T_{n_{\mathrm{cal}}+1} = \infty$. The calibration threshold $\hat{\tau}$ is then set as

$$\hat{\tau} := T_{(k)}, \qquad k = \lceil (n_{cal} + 1)(1 - \alpha) \rceil .$$

This is the k-th order statistic of $\{T_i\}_{i=1}^{n_{\mathrm{cal}}+1}$. Then, the distribution of C-USIM's conditional coverage C can be written as $C = P_{\hat{\tau}}(X)$.

Under Assumptions 2 and 3, if $n_{\mathrm{cal}} \to \infty$, the threshold $\hat{\tau}$ converges in probability to $q_{1-\alpha}$, which is the $(1 - \alpha)$ quantile of the marginal score distribution, and $P_{\hat{\tau}}(X)$ converges in distribution to $P_{q_{1-\alpha}}(X)$. The ideal case for C-USIM is as follows.

Theorem 2 (Conditional coverage distribution in the ideal case)**.** *Under Assumptions 2 and 3, suppose that $\alpha > (n_{\mathrm{cal}} + 1)^{-1}$ and $T(X, Y) = f(P(X, Y))$ for some strictly increasing function f almost surely. Then C-USIM's conditional coverage C follows* $\mathrm{Beta}(k, n_{\mathrm{cal}} + 1 - k)$.

For each test covariate X_i, consider the percentile rank–score curve $(P(X_i, y), T(X_i, y))$. For a given threshold t, $P_t(X_i)$ is the horizontal coordinate at which this curve intersects the horizontal line $T = t$. More generally, $P_t(X)$ describes the conditional-coverage distribution at threshold t. Figure 2 illustrates these percentile rank–score plots (a) and the resulting conditional-coverage distributions (b) in an example setting.

The diagonal $T = P$ indicates agreement between the score and its conditional percentile rank given $X = x$. At the nominal threshold, curves above (below) the diagonal indicate undercoverage (or overcoverage). For fixed predictive distributions, the calibration shifts the horizontal line while leaving the curves unchanged. In this example, C-USIM moves this line toward $q_{1-\alpha} \approx 0.973$. If the model satisfies Assumption 2, this is the point to which its calibrated threshold converges as the calibration sample size increases. Meanwhile, plug-in HPD uses the fixed score threshold 0.95. Figure 2(a) shows the intersections, and Figure 2(b) summarizes their horizontal coordinates ($P_{0.973}(X_i)$ for ideal C-USIM and $P_{0.95}(X_i)$ for plug-in HPD).

This representation also explains the roles of training and calibration data. Increasing the training sample aims to improve the predictive distributions and bring the curves closer to $T = P$, whereas more calibration data can stabilize the estimated threshold. For TabPFN and TabICL, we observed diminishing gains beyond a certain training size. In this regime, reallocating observations from training to calibration improved coverage more than further enlarging the training set. More illustrations of percentile rank–score plots and conditional coverage distributions using TabPFN and TabICL are attached in Appendix B.3. Section 6.3 examines this trade-off under a fixed label budget.

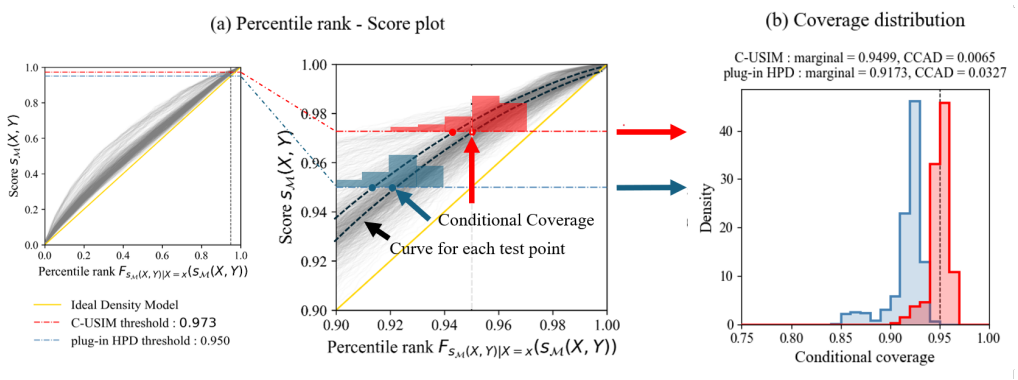

Figure 2: Percentile rank–score plots over test covariates (plot (a)), obtained using TabPFN on the sine with shifted exponential noise function (1D-1). Plot (b) approximates $P_t(X_i)$ at the nominal score threshold $t = 1-\alpha$ (blue) and the ideal C-USIM threshold $t = q_{1-\alpha}$ (red), which is estimated by the empirical $(1-\alpha)$-quantile of the pooled Monte Carlo scores. The histogram summarizes interpolated estimates of $P_t(X_1), \ldots, P_t(X_{n_{test}})$, approximating the distribution of $P_t(X)$. See Appendices B.2 and B.3 for details.

6 Experiments

We compare C-USIM with plug-in HPD on synthetic and real-world regression tasks, evaluating coverage accuracy and prediction-set length.

Calibrating method. Throughout all experiments, we obtain distribution estimates for the calibration and test covariates by providing the row-merged dataset $D_{\text{cal}} \dot{\cup} D_{\text{test}}$, following Algorithm 1. Because TabPFN and TabICL produce outputs in different formats, we use model-specific reconstruction rules to convert them into the finite piecewise-uniform conditional densities described in Appendix B.1. For the plug-in HPD baseline, we include the entire boundary density level required to attain at least 95% model mass, following Section 2.2. In contrast, C-USIM uses the exact score sublevel set determined by its calibration order statistic, following Algorithm 1. Appendix B.5 provides the common settings, random seeds, and grouping scheme.

Evaluation measures. We report marginal coverage and total prediction-set length alongside conditional coverage absolute deviation (CCAD) for synthetic data and CEC-X for real data. For synthetic data with known true conditional densities, CCAD measures mean absolute conditional coverage error across inputs. For real data, where these densities are unknown, CEC-X averages absolute empirical coverage errors across covariate-defined groups, weighted by their test sizes. These measures correspond to the three goals in Section 4.2: marginal validity, conditional coverage close to the target, and small prediction sets. Appendix B.4 provides the formal definitions for these.

Fixed-context comparison. Our main comparison uses a total budget of 1,536 labels: plug-in HPD uses all labels as context, while C-USIM uses 512 for context and 1,024 for calibration. To examine the effect of calibrating the cutoff with the predictive densities held fixed, we also compare C-USIM with plug-in HPD using the same 512-observation context. Only C-USIM uses the additional 1,024 calibration responses in this comparison. We report this fixed-context comparison as a supplementary analysis in Appendix B.8, since our primary comparison concerns context–calibration allocation under a common total label budget.

6.1 Function-based synthetic examples

We compare plug-in HPD and C-USIM using the settings in Table 1 in Appendix B.5. We consider three one-dimensional and three multidimensional synthetic examples, described in Appendix B.2. We selected these examples to assess coverage in nonlinear settings with oscillatory or discontinuous response functions, skewed noise, and covariate-dependent scale mixtures. For each model, we report averages across all six mechanisms.

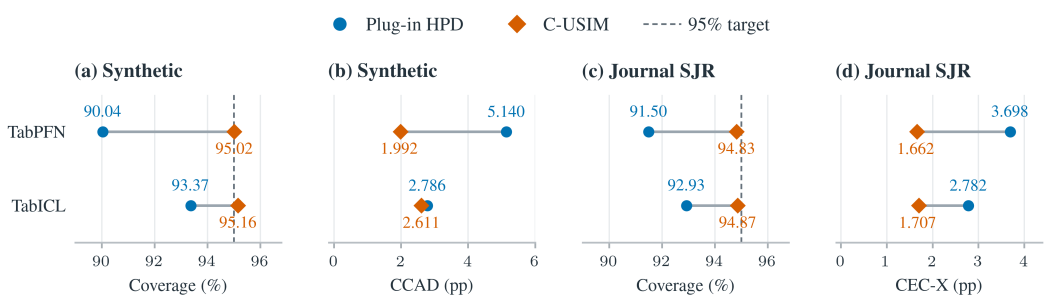

Figure 3: Synthetic and Journal SJR overview. (a,b) Synthetic seed means, averaged equally over the selected mechanisms. (c,d) Journal SJR seed means of marginal coverage and CEC-X, respectively; (d) uses a representative covariate grouping. See Table 1 for experiment settings and Appendix B.6 for detailed results.

Figure 3(a,b) summarizes results from the six selected mechanisms. C-USIM brings mean marginal coverage closer to the target and reduces mean CCAD for both models, with a larger reduction for TabPFN. The lower mean CCAD shows that the gains extend to average conditional coverage accuracy across these mechanisms. Under the same total label budget, these results support reserving labels for calibration when constructing prediction regions from TFM outputs. Appendix B.6.1 gives the case-specific seed distributions, numerical results, and set lengths.

6.2 Real-world regression examples

We evaluate C-USIM on three real-world datasets: Journal SJR, JP Anime, and Allstate Claims Severity. Journal SJR, from the CARTE benchmark (Kim et al., 2024), contains publication metadata and journal H-indices from SCImago.[3] Descriptions and results for the other two datasets are provided in Appendix B.6.3. We use K-means to construct multiple covariate groupings for CEC-X evaluation. Detailed settings and grouping procedures are provided in Table 1 and Appendix B.5.

Figure 3(c,d) shows that C-USIM brings mean marginal coverage closer to the target and lowers CEC-X for both models. Mean group error also decreases for both models. With the same total label budget as plug-in HPD, C-USIM thus improves both marginal coverage accuracy and average coverage accuracy within groups. Appendix B.6.2 provides detailed group results and comparisons.

6.3 Split ratio

We examine how the training–calibration split affects point-prediction accuracy and coverage stability in C-USIM. An 8:2 training–calibration split is a customary starting point for point prediction, but Das et al. (2026) show that the preferred split for prediction-interval length depends on the learning regime. To assess coverage stability, we use the rank–score representation in Section 5.2: training context shapes the input-specific curves, whereas, for a fixed predictor, more calibration data can stabilize the horizontal line $T = \hat{\tau}$. When additional training yields little improvement in the curves' concentration around $T = P$, allocating more labels to calibration may improve coverage stability.

Figure 4 shows how the training–calibration allocation affects coverage stability for C-USIM on MD-2 from Section 6.1, using the fixed label budget and settings in Table 1. The first two columns show clear trade-offs between RMSE and absolute coverage error, with a smaller calibration size leading to better point prediction accuracy while increasing the absolute coverage error. For TabPFN, a 2:1 or 1:1 split seems to strike a good balance between prediction accuracy and coverage stability. For TabICL, there is no such balancing point, showing clear trade-offs between the two goals. CCAD also increases from the 1:2 split through 5:1. The reduction in marginal coverage error holds across all evaluated case–model pairs (Appendix B.7), supporting larger calibration sets when stable coverage is a priority.

[3]Dataset source: `https://huggingface.co/datasets/inria-soda/carte-benchmark`.

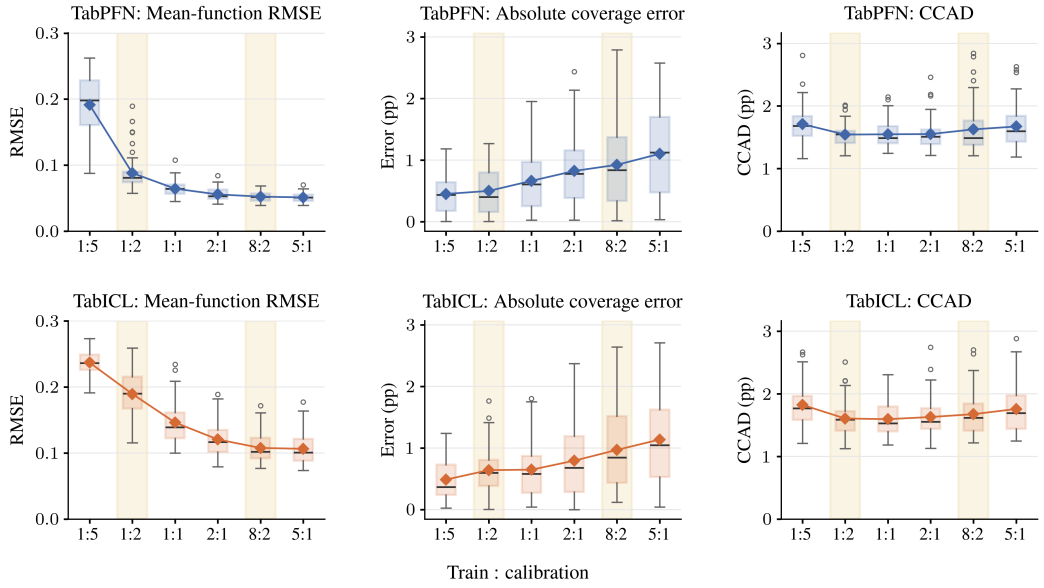

Figure 4: Split-ratio sensitivity on MD-2; settings follow Table 1. Rows correspond to TabPFN and TabICL. Columns show mean-function RMSE, absolute marginal coverage error, and CCAD; lower values are better. Standard boxplots summarize variation across seeds; connected diamonds mark the means.

7 Limitations and Future Work

The distribution estimates produced by current TFMs may depend on internal model settings, the calibration and test covariates jointly processed in a prediction call, and the ordering of the training rows; Assumption 2 rules out the dependence on the other jointly processed test covariates. Although these factors do not necessarily violate the exchangeability assumption required for conformal prediction, they make it difficult to characterize the generalized behavior of conditional coverage distributions of C-USIM.

C-USIM can be extended to other scoring criteria by computing model-based PIT transformations from the TFM predictive distribution. For instance, if the conditional response distribution is symmetric, a PIT-corrected symmetric density score ($s(x, y) = \mathbb{P}_{Z\sim\mathcal{M}(x)}(f_{\mathcal{M}(x)}(y) + f_{\mathcal{M}(x)}(2\hat{y}^{\text{med}} - y) \le f_{\mathcal{M}(x)}(Z) + f_{\mathcal{M}(x)}(2\hat{y}^{\text{med}} - Z))$), where $\hat{y}^{\text{med}}$ is the estimated median response, may be considered. Future work could establish other generalized pivotal scores and evaluate their conditional coverage and efficiency on synthetic and real-world data.

8 Conclusion

C-USIM provides lightweight HPD-split calibration with marginal validity under the stated exchangeability assumptions. It accommodates multimodal predictions through potentially disjoint regions, using pretrained TFM calibration and test outputs without further training or inference. Our analysis bounds the conditional coverage gap and uses percentile rank–score plots to characterize how calibration thresholds affect coverage across inputs. Experiments with TabPFN and TabICL show that, under a fixed label budget, C-USIM improves marginal coverage accuracy over plug-in HPD, reduces CCAD in most evaluated synthetic cases, and lowers aggregate group coverage errors in most comparisons on real data cases. The split-ratio results further suggest that allocations for conformal prediction may differ from the training-heavy splits motivated by point prediction.

AI USE STATEMENT

In this work, we used generative AI tools for implementing methods. We have not used generative AI tools for proposing or refining hypotheses, interpreting results, or constructing mathematical proofs, and qualitative and thematic data analysis is not applicable to this work. Additionally, we used generative AI tools for creating or editing software code, creating or modifying scientific figures, editing the paper to improve readability, and checking mathematical proofs. We have reviewed all AI-assisted work. In particular, we checked the correctness of the AI-generated code. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

Code and detailed configurations will be released on GitHub upon publication. Algorithm 1 specifies the C-USIM procedure, and Appendix A provides proofs of the theoretical results under the stated assumptions. Appendix B documents how predictive densities are constructed from TFM outputs, the synthetic data-generating processes, and the evaluation measures. The experimental settings in Appendices B.5 and B.6.3 include sample sizes, random seeds, training–calibration allocations, and preprocessing and group construction for the real-world datasets. Appendix B.7 gives further details of the allocation experiments.

REFERENCES

Rina Foygel Barber, Emmanuel J. Candès, Aaditya Ramdas, and Ryan J. Tibshirani. The limits of distribution-free conditional predictive inference. *Information and Inference: A Journal of the IMA*, 10(2):455–482, 2021. doi: 10.1093/imaiai/iaaa017. URL https://arxiv.org/abs/1903.04684.

José Lucas De Melo Costa, Fabrice Popineau, Arpad Rimmel, and Bich-Liên Doan. High performance, low reliability: Uncertainty benchmarking for tabular foundation models. In *Proceedings of the European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning*, pp. 115–120. i6doc.com, 2026. doi: 10.14428/esann/2026.ES2026-261. URL https://arxiv.org/abs/2605.28554.

Sayan Das, Bahram Yaghooti, Todd A Kuffner, and Soumendra N Lahiri. On optimal data splitting for split conformal prediction. *arXiv:2606.31600*, 2026. URL https://arxiv.org/abs/2606.31600.

Victor Dheur, Matteo Fontana, Yorick Estievenart, Naomi Desobry, and Souhaib Ben Taieb. A unified comparative study with generalized conformity scores for multi-output conformal regression. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 13444–13485. PMLR, 2025. URL https://arxiv.org/abs/2501.10533.

Eshant English and Christoph Lippert. Japan: Joint adaptive prediction areas with normalising flow. In *International Conference on Learning Representations*, volume 2026, pp. 131010–131047, 2026.

A Gammerman, V Vovk, and V Vapnik. Learning by transduction, vol uai'98, 1998.

Léo Grinsztajn, Klemens Flöge, Oscar Key, Felix Birkel, Philipp Jund, Brendan Roof, Mihir Manium, Shi Bin Hoo, Magnus Bühler, Anurag Garg, Dominik Safaric, Jake Robertson, Benjamin Jäger, Simone Alessi, Adrian Hayler, Vladyslav Moroshan, Lennart Purucker, Philipp Singer, Alan Arazi, Julien Siems, Jan Hendrik Metzen, Georg Grab, Nick Erickson, Siyuan Guo, Eliott Kalfon, Simon Bing, David Salinas, Clara Cornu, Lilly Charlotte Wehrhahn, Diana Kriuchkova, Kursat Kaya, Lydia Sidhoum, Marie Salmon, Jerry Chen, Madelon Hulsebos, Yann LeCun, Samuel Müller, Bernhard Schölkopf, Sauraj Gambhir, Noah Hollmann, and Frank Hutter. TabPFN-3: Technical report. *arXiv:2605.13986*, 2026. URL https://arxiv.org/abs/2605.13986.

Rob J. Hyndman. Computing and graphing highest density regions. *The American Statistician*, 50 (2):120–126, 1996. URL https://robjhyndman.com/papers/hdrgraph.pdf.

Rafael Izbicki, Gilson Shimizu, and Rafael B. Stern. Cd-split and hpd-split: efficient conformal regions in high dimensions. *Journal of Machine Learning Research*, 23(1), January 2022. ISSN 1532-4435.

Patrik Kenfack, Samira Ebrahimi Kahou, and Ulrich Aïvodji. Towards fair in-context learning with tabular foundation models. *Transactions on Machine Learning Research*, 2026. URL https://arxiv.org/abs/2505.09503.

Myung Jun Kim, Leo Grinsztajn, and Gael Varoquaux. CARTE: Pretraining and transfer for tabular learning. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 23843–23866. PMLR, 2024. URL https://proceedings.mlr.press/v235/kim24d.html.

Félix Laplante. A post-processing conformal prediction approach for conditional coverage via pivotal scores. *arXiv:2605.25852*, 2026. URL https://arxiv.org/abs/2605.25852.

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

Rui Luo and Zhixin Zhou. Density-sorted prediction set: Efficient conformal prediction for multi-target regression. *Pattern Recognition*, pp. 112513, 2025.

Samuel Müller, Noah Hollmann, and Frank Hutter. Bayes' power for explaining in-context learning generalizations. *arXiv:2410.01565*, 2024. URL https://arxiv.org/abs/2410.01565.

Jingang Qu, David Holzmüller, Gaël Varoquaux, and Marine Le Morvan. TabICLv2: A better, faster, scalable, and open tabular foundation model. In *International Conference on Machine Learning*, 2026. URL https://arxiv.org/abs/2602.11139.

Yaniv Romano, Evan Patterson, and Emmanuel J. Candès. Conformalized quantile regression, 2019. URL https://arxiv.org/abs/1905.03222.

Craig Saunders, Alex Gammerman, and Volodya Vovk. Transduction with confidence and credibility. In *Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence*, volume 2, pp. 722–726. Morgan Kaufmann, 1999.

Matteo Sesia and Yaniv Romano. Conformal prediction using conditional histograms, 2021. URL https://arxiv.org/abs/2105.08747.

Glenn Shafer and Vladimir Vovk. A tutorial on conformal prediction. *Journal of Machine Learning Research*, 9(3), 2008.

Florian D. van Leeuwen. Conformal prediction for tabular prior-data fitted networks with missing data. In *EurIPS 2025 Workshop: AI for Tabular Data*, 2025. URL https://openreview.net/forum?id=TnZcC7GXI5.

Vladimir Vovk, Alex Gammerman, and Glenn Shafer. *Algorithmic Learning in a Random World*. Springer, New York, NY, 2005. doi: 10.1007/b106715.

A PROOFS

A.1 PROOF OF THEOREM 1

Proof. The quantity $\Delta_{\tilde{s}}(x)$ is bounded by the Kolmogorov–Smirnov (KS) distance between the conditional and marginal score distributions (Laplante, 2026):

$$\Delta(x)_{\tilde{s}} \leq d_{KS}(F_{\tilde{s}|X=x}, F_{\tilde{s}})$$

Let $(t_1^{(x)}, \ldots, t_{r+1}^{(x)})$ be a permutation of $(1, \ldots, r+1)$ such that

$$S^{(x)}_{t_i^{(x)}} \leq S^{(x)}_{t_j^{(x)}} \qquad \text{for } i < j.$$

The probability measure of the PIT-corrected score $\tilde{s}$ is:

$$\mu_{\tilde{s}|X=x} = \sum_{i=1}^{r+1} \tilde{p}^{(x)}_{t_i} \delta_{\sum_{j=1}^{i} \hat{p}^{(x)}_{t_j}}$$

$$\begin{aligned} d_{KS}(F_{\tilde{s}|X=x}, \mathrm{Unif}[0,1]) &= \max_{i \in [r+1]} \max \left(\left| \sum_{j=1}^{i} (\tilde{p}^{(x)}_{t_j} - \hat{p}^{(x)}_{t_j}) \right|, \left| \sum_{j=1}^{i-1} \tilde{p}^{(x)}_{t_j} - \sum_{j=1}^{i} \hat{p}^{(x)}_{t_j} \right| \right) \\ &\leq \frac{1}{2} \|\tilde{p}^{(x)} - \hat{p}^{(x)}\|_1 + \max_{i \in [r+1]} \hat{p}^{(x)}_i = B(x) \end{aligned}$$

By the convexity of the KS distance, and since $\mu_{\tilde{s}}$ is the expectation of the conditional distribution $\mu_{\tilde{s}|X=x}$, the inequality below holds.

$$\begin{aligned} d_{\mathrm{KS}}(F_{\tilde{s}}, \mathrm{Unif}(0,1)) &\leq \mathbb{E}_X \left[d_{\mathrm{KS}}(F_{\tilde{s}|X}, \mathrm{Unif}(0,1)) \right] \\ &\leq \mathbb{E}_X[B(X)] \end{aligned}$$

Therefore, by the triangle inequality of the KS distance,

$$\begin{aligned} d_{KS}(F_{\tilde{s}|X=x}, F_{\tilde{s}}) &\leq d_{KS}(F_{\tilde{s}}, \mathrm{Unif}[0,1]) + d_{KS}(F_{\tilde{s}|X=x}, \mathrm{Unif}[0,1]) \\ &\leq \mathbb{E}_X[B(X)] + B(x). \end{aligned}$$

□

A.2 PROOF OF THEOREM 2

Proof. The theorem's assumption on the strictly increasing function f implies that $f^{-1} = F_{s_{\mathcal{M}}(X,Y)|X=x}$ almost surely regardless of x. Thus,

$$\begin{aligned} P_t(x) &= \mathbb{P}\left(s_{\mathcal{M}}(X,Y) \leq t \mid X = x\right) \\ &= \mathbb{E}[\mathbb{P}\left(s_{\mathcal{M}}(X,Y) \leq t \mid X = x\right)] \\ &= P\left(s_{\mathcal{M}}(X,Y) \leq t\right) \\ &= F_{s_{\mathcal{M}}(X,Y)}(t) \end{aligned}$$

Because of the assumption, $F_{s_{\mathcal{M}}(X,Y)}(s_{\mathcal{M}}(X,Y)) \sim \mathrm{Unif}(0,1)$, and since a cumulative distribution function preserves order, $C = P_{\hat{\tau}}(X) = F_{s_{\mathcal{M}}(X,Y)}(\hat{\tau})$ has the distribution of the k-th order statistic of n_{cal} independent $\mathrm{Unif}(0,1)$ random variables, where $k = \lceil (n_{\mathrm{cal}}+1)(1-\alpha) \rceil \leq n_{\mathrm{cal}}$. This order statistic follows $\mathrm{Beta}(k, n_{\mathrm{cal}} + 1 - k)$. □

B EXPERIMENTAL DETAILS

B.1 DENSITY RECONSTRUCTION FROM TFM OUTPUTS

The following reconstruction rules convert TFM outputs into piecewise-constant densities without fitting a separate density estimator.

Binning-based Outputs. For binning-based models, we represent the estimated conditional distribution at x by contiguous predicted intervals $\{[\hat{y}_{i-1}^{(x)}, \hat{y}_i^{(x)})\}_{i=1}^{r+1}$, where $\hat{y}_0^{(x)} < \hat{y}_1^{(x)} < ... < \hat{y}_{r+1}^{(x)}$ and their corresponding probability weights $\{\hat{p}_i^{(x)}\}_{i=1}^{r+1}$. We construct a piecewise-constant density estimate by dividing each predicted probability weight by the width of its corresponding interval. Specifically,

$$f_{\mathcal{M}(x)}(y) = \sum_{i=1}^{r+1} \frac{\hat{p}_i^{(x)}}{\hat{y}_i^{(x)} - \hat{y}_{i-1}^{(x)}} \mathbf{1}\left\{y \in [\hat{y}_{i-1}^{(x)}, \hat{y}_i^{(x)})\right\}.$$

For TabPFN, we consider the default setting ($r = 4999$, $\hat{y}_i^{(x)}$ is fixed with respect to x) (Grinsztajn et al., 2026).

Quantile-based Outputs. For quantile-based models, we represent the estimated conditional distribution at x by the set of quantile level-value pairs $\{(q_i^{(x)}, y_i^{(x)})\}_{i=0}^{r+1}$ such that $0 = \hat{q}_0^{(x)} < \hat{q}_1^{(x)} < \hat{q}_2^{(x)} < ... < \hat{q}_{r+1}^{(x)} = 1$, $\hat{y}_0^{(x)} < \hat{y}_1^{(x)} < \hat{y}_2^{(x)} < ... < \hat{y}_{r+1}^{(x)}$. We construct a piecewise-constant density estimate by dividing each percentile gap by the gap between quantiles. Specifically,

$$f_{\mathcal{M}(x)}(y) = \sum_{i=1}^{r+1} \frac{\hat{q}_i^{(x)} - \hat{q}_{i-1}^{(x)}}{\hat{y}_i^{(x)} - \hat{y}_{i-1}^{(x)}} \mathbf{1}\left\{y \in [\hat{y}_{i-1}^{(x)}, \hat{y}_i^{(x)})\right\}.$$

For TabICL, we use the default quantile grid $\left(r = 999, \hat{q}_i^{(x)} = \frac{i}{r+1}\right)$. The model does not predict the boundary quantiles $\hat{y}_0^{(x)}, \hat{y}_{r+1}^{(x)}$ (Qu et al., 2026). Before constructing the density, we retain strictly increasing raw quantile sequences and repair only those with crossings or duplicate values, using float64 arithmetic. For a raw sequence $v_0, \ldots, v_{998}$, let R be its range, replaced by $\max(1, |v_0|)$ if the sequence is constant, and set $g = \max\{10^{-6}R/998,\ 8\,\mathrm{ulp}(\max_j |v_j|)\}$, where ulp denotes the spacing to the next larger float64 value. We apply unweighted least-squares isotonic regression to $v_j - jg$ using the pool-adjacent-violators algorithm and then restore the ramp jg. The resulting strictly increasing sequence supplies the interior quantiles in the density above. After this correction, we set the boundary quantile values to $\hat{y}_1^{(x)} - P \cdot (\hat{y}_2^{(x)} - \hat{y}_1^{(x)})$ and $\hat{y}_r^{(x)} + P \cdot (\hat{y}_r^{(x)} - \hat{y}_{r-1}^{(x)})$, respectively, with P fixed at 3 throughout our experiments. This gives 1,000 finite intervals, each with probability mass 0.001.

B.2 Synthetic data-generating processes

Within each example, context, calibration, and test observations (and hence their covariates) are sampled i.i.d. from the same joint distribution.

B.2.1 One-dimensional Examples

The three one-dimensional examples use $X \sim \mathrm{Unif}(0, 2\pi)$. Throughout this appendix, $Z \sim \mathcal{N}(0, 1)$ is independent of the covariates and any mixture-component draw. All Gaussian scale parameters below are standard deviations.

1D-1: Sine with shifted exponential noise. For $E \sim \mathrm{Exp}(1)$ independent of X, the response is

$$Y = \sin(10X) + 0.18(E - 1).$$

The exponential distribution has rate one, so the noise is centered and has variance 0.18^2.

1D-2: Periodic Gaussian scale mixture. Given $X = x$, draw $B \sim \mathrm{Bernoulli}(p(x))$, independently of Z, where

$$p(x) = 0.08 + 0.30(0.5 + 0.5\sin(5x)).$$

Then

$$Y = 0.8\sin(12X) + 0.2\cos(3X) + \sigma_B Z, \qquad \sigma_B = \begin{cases} 0.48, & B = 1, \\ 0.07, & B = 0. \end{cases}$$

Thus, $p(x)$ is the probability of the wider Gaussian component.

1D-3: Step function. The response is

$$Y = m(X) + 0.13Z, \qquad m(x) = \begin{cases} 0.85, & \sin(10x) \geq 0, \\ -0.85, & \sin(10x) < 0. \end{cases}$$

B.2.2 MULTI-DIMENSIONAL EXAMPLES MD-1 – MD-3

The three multi-dimensional examples use i.i.d. coordinates $X_j \sim \mathrm{Unif}(0,1)$, $j = 1, \ldots, d$. These examples share the mean

$$m(x) = \sin(2\pi x_1) + 0.65(2x_2 - 1)(2x_3 - 1).$$

Given $X = x$, draw $B \sim \mathrm{Bernoulli}(p(x))$ independently of $Z \sim \mathcal{N}(0,1)$ and set $Y = m(X) + \sigma_B(X)Z$. The parameters below specify the probability and standard deviation of each mixture component. These mechanisms were selected during earlier exploratory screening; the displayed evaluation includes these three mechanisms for both models.

MD-1: Input-dependent severity, $d = 5$. The mixture probability is $p(x) = 0.20$, with $\sigma_0(x) = 0.05$ and $\sigma_1(x) = 0.30 + 0.90x_4$. The fourth coordinate changes the scale of the wider component.

MD-2: Radial mixture weight, $d = 10$. Define

$$r(x) = \left\{ \frac{1}{7} \sum_{j=4}^{10} (2x_j - 1)^2 \right\}^{1/2}, \qquad p(x) = 0.03 + 0.65 \,\mathrm{clip}\left(\frac{r(x) - 0.25}{0.60}, 0, 1 \right),$$

where $\mathrm{clip}(a, 0, 1) = \min(1, \max(0, a))$. The component scales are $\sigma_0 = 0.05$ and $\sigma_1 = 1.00$.

MD-3: Input-dependent scale mixture, $d = 20$. Here $p(x) = 0.05 + 0.50x_4$, $\sigma_0 = 0.07$, and $\sigma_1 = 0.90$. The remaining 16 coordinates do not affect the conditional response law.

B.3 CONSTRUCTION OF PERCENTILE RANK–SCORE PLOTS AND CONDITIONAL COVERAGE DISTRIBUTIONS

First, for each fixed X_i, we independently draw two sets of Monte Carlo samples:

$$\widetilde{Y}_i^{(1)}, \ldots, \widetilde{Y}_i^{(B)} \overset{\text{i.i.d.}}{\sim} Y \mid X = X_i, \qquad Y_i^{(1)}, \ldots, Y_i^{(L)} \overset{\text{i.i.d.}}{\sim} Y \mid X = X_i.$$

The first set serves as a reference sample for approximating the conditional percentile rank:

$$\widehat{P}(X_i, y) := \frac{1}{B} \sum_{b=1}^{B} \mathbf{1}\left\{ T(X_i, \widetilde{Y}_i^{(b)}) \leq T(X_i, y) \right\}.$$

For each X_i, we plot an empirical percentile rank–score curve by sorting and connecting the L points

$$\left\{ \widehat{P}(X_i, Y_i^{(\ell)}), T(X_i, Y_i^{(\ell)}) \right\}_{\ell=1}^{L}$$

We overlay these curves for $i = 1, \ldots, m$. The conditional-coverage distribution is approximated from the horizontal coordinates of the intersections between these interpolated curves and the horizontal line at each score threshold. We approximate $q_{1-\alpha}$ by the empirical $(1-\alpha)$-quantile of the pooled Monte Carlo scores.

Under Assumptions 2 and 3, the population threshold satisfies $\mathbb{E}[P_{q_{1-\alpha}}(X)] = \mathbb{P}[s_{\mathcal{M}}(X, Y) \leq q_{1-\alpha}] = 1 - \alpha$. For a discrete marginal score law, coverage at this threshold is at least $1 - \alpha$ and can exceed it because of a jump in the score CDF.

Generally, the predictive distributions of TabPFN and TabICL in our setting are piecewise uniform, so Assumption 3 does not hold exactly.

In particular, the identity in Section 5.2,

$$P(x, y) = \mathbb{P}(s_{\mathcal{M}}(x, Z) < s_{\mathcal{M}}(x, y) \mid X = x)$$

need not hold rigorously, since the inequality

$$\mathbb{P}\Big(Z \in \widehat{C}_{T(x,y)}(x) \mid X = x\Big) \geq \mathbb{P}(s_{\mathcal{M}}(x, Z) < s_{\mathcal{M}}(x, y) \mid X = x)$$

is strict whenever that level has positive probability under the true conditional response law; this probability is the difference between the two sides. Discrete scores do not satisfy exact PIT uniformity but both models yield sufficiently fine discretizations that this discrepancy can reasonably be regarded as negligible and as having no material effect on the overall patterns in the plots.

Unless otherwise noted, the top two rows of each figure show TabPFN, and the bottom two rows show TabICL. Rows with $n_{\text{train}} = 4096$ alternate with rows with $n_{\text{train}} = 1024$, starting from the top. We set $B = L = 10000$. Plot (a) shows percentile rank–score plots, whereas plot (b) shows the interpolated conditional-coverage distributions at ideal C-USIM and plug-in HPD.

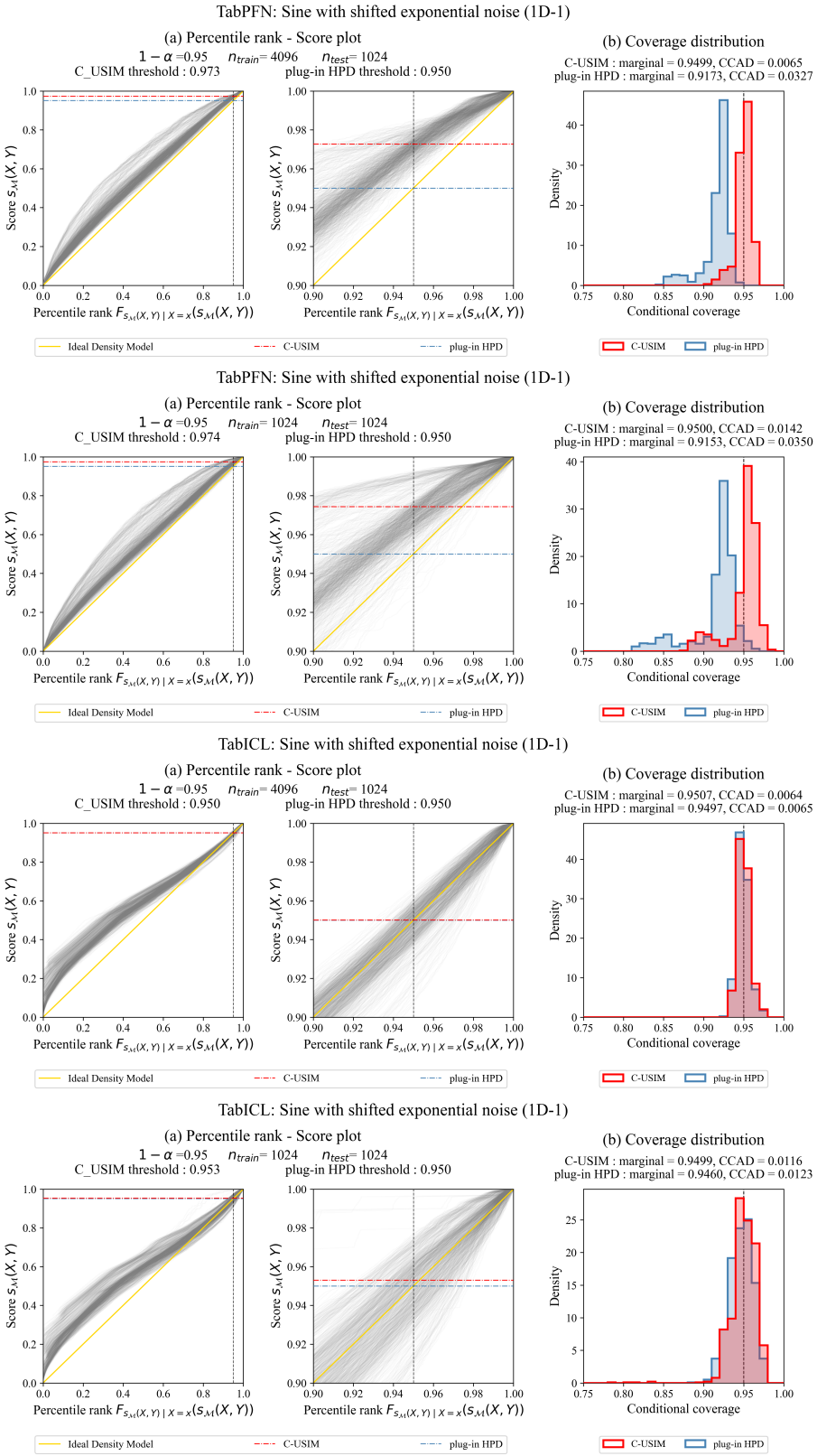

Figure 5: Percentile rank–score diagnostics for 1D-1, seed 2026: C-USIM ($n_{\mathrm{cal}} \to \infty$) and plug-in HPD.

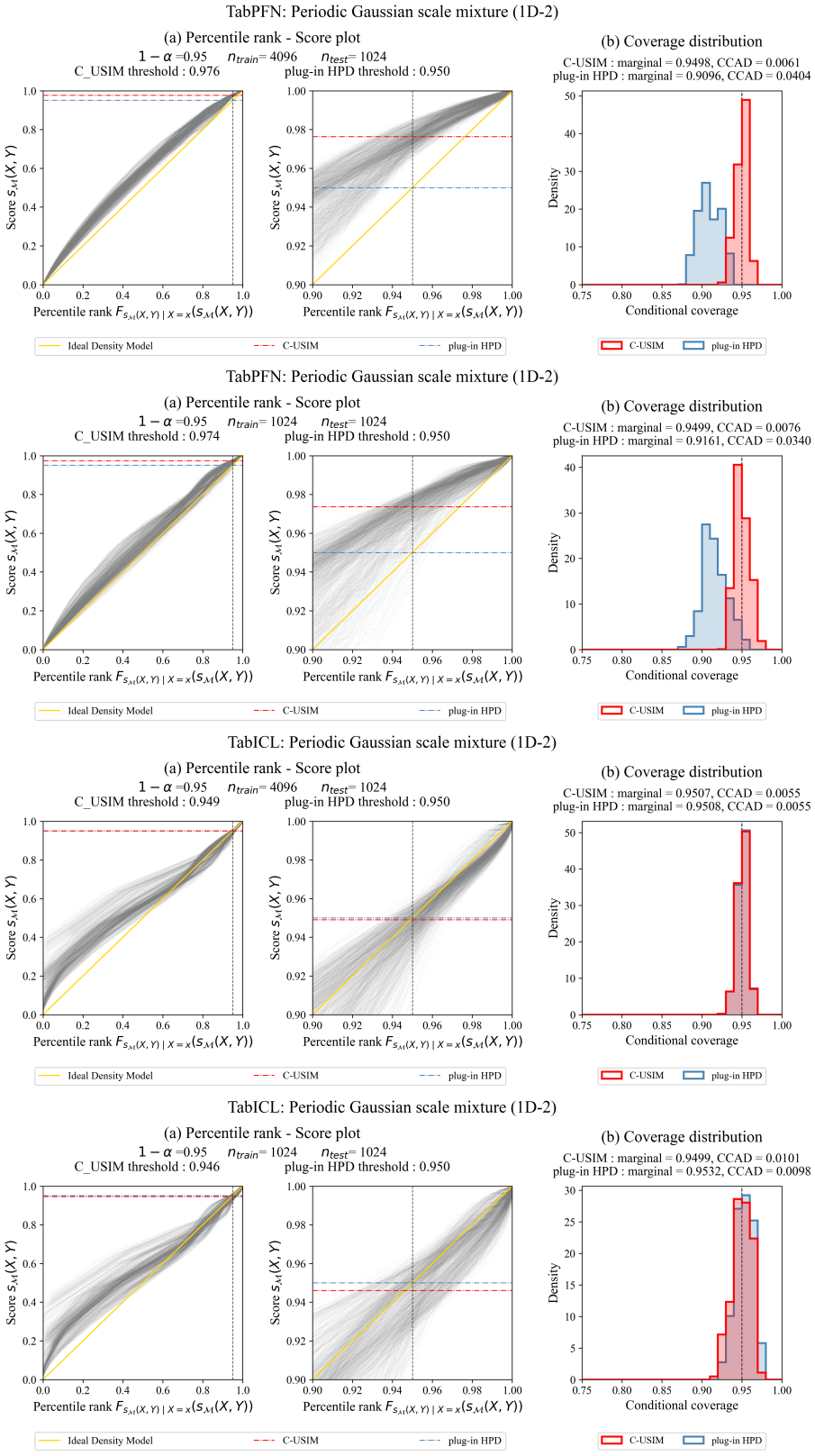

Figure 6: Percentile rank–score diagnostics for 1D-2, seed 2026: C-USIM ($n_{\mathrm{cal}} \to \infty$) and plug-in HPD.

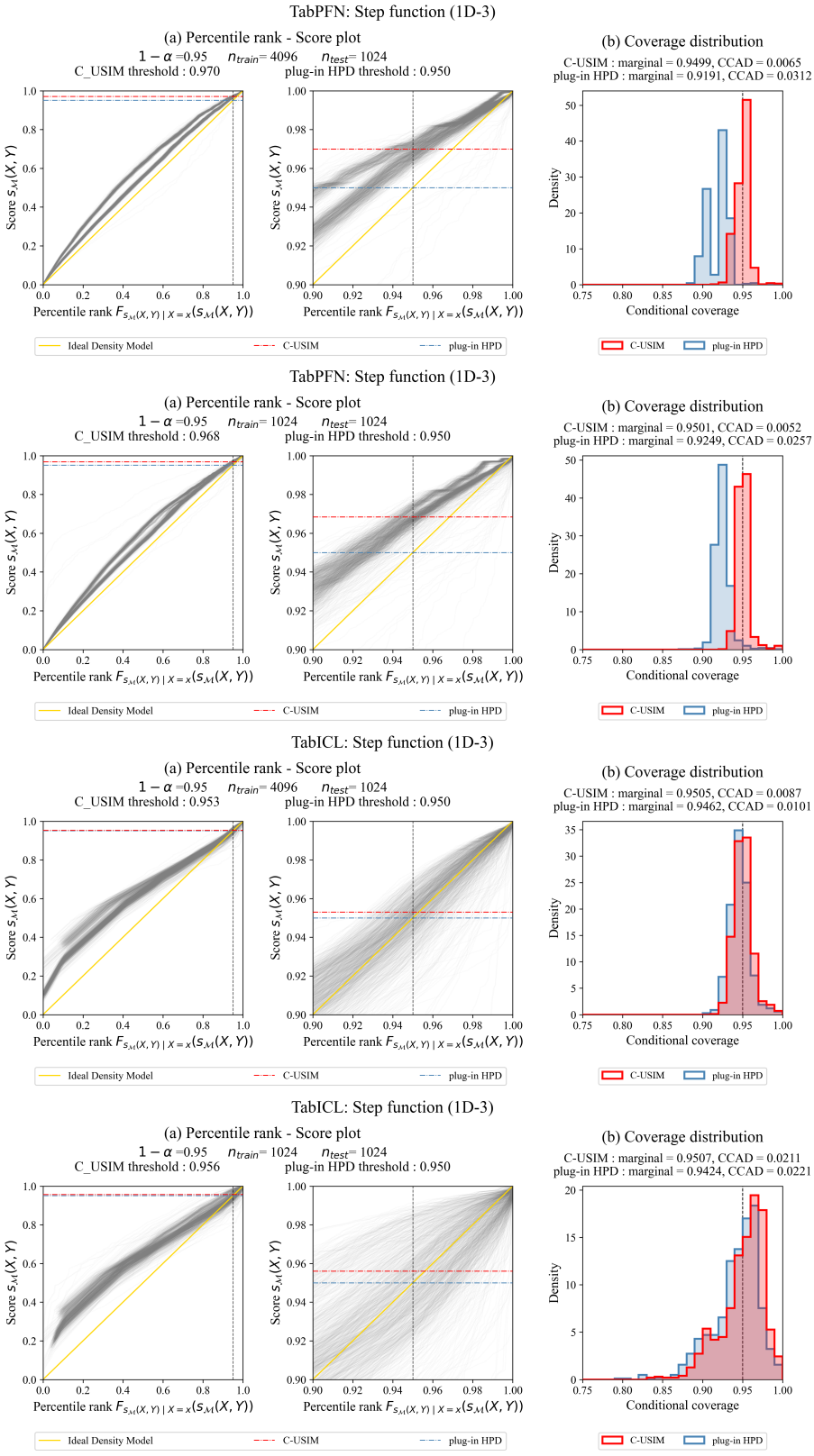

Figure 7: Percentile rank–score diagnostics for 1D-3, seed 2026: C-USIM ($n_{\mathrm{cal}} \to \infty$) and plug-in HPD.

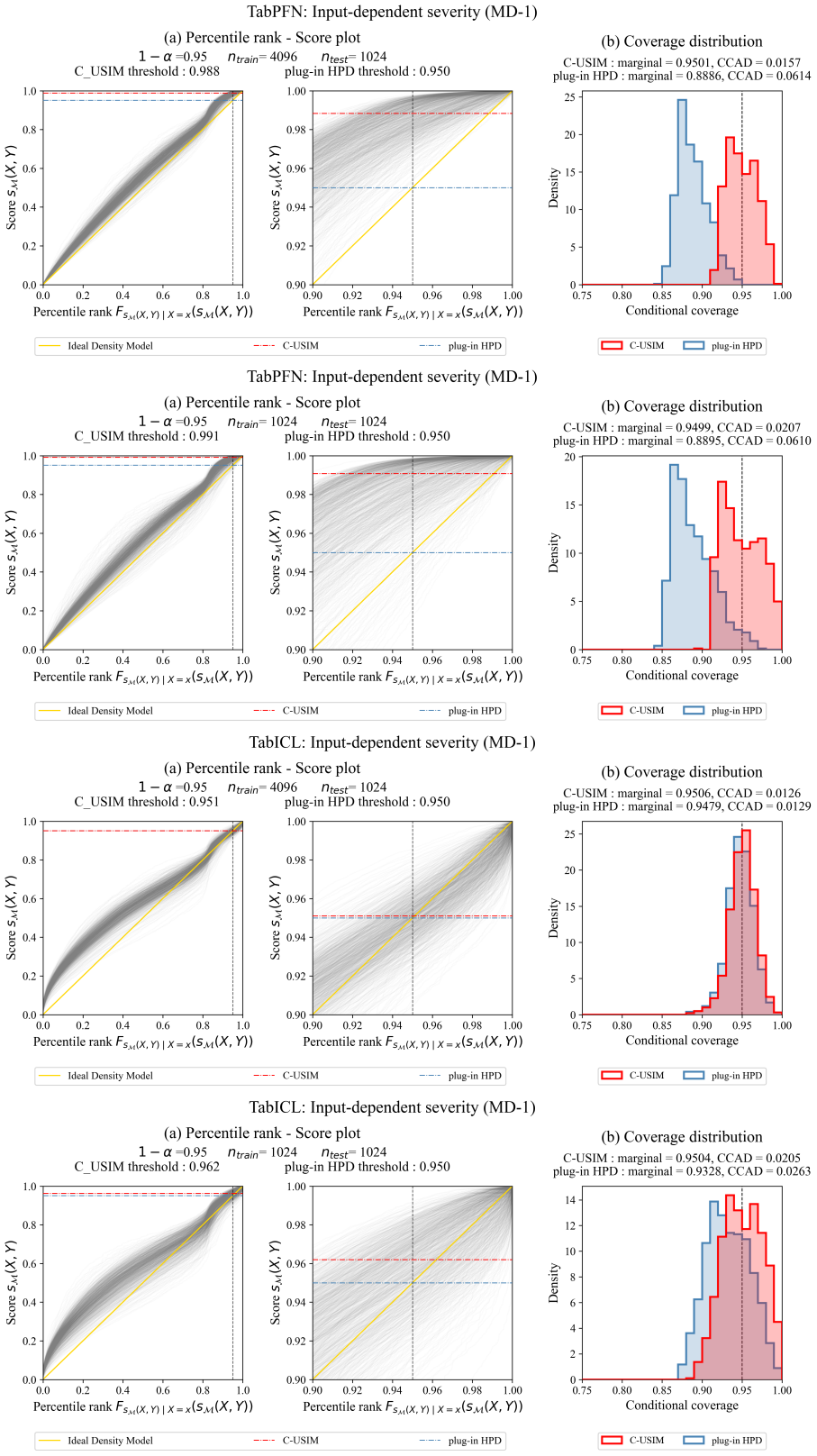

Figure 8: Percentile rank–score diagnostics for MD-1, seed 2026: C-USIM ($n_{\mathrm{cal}} \to \infty$) and plug-in HPD.

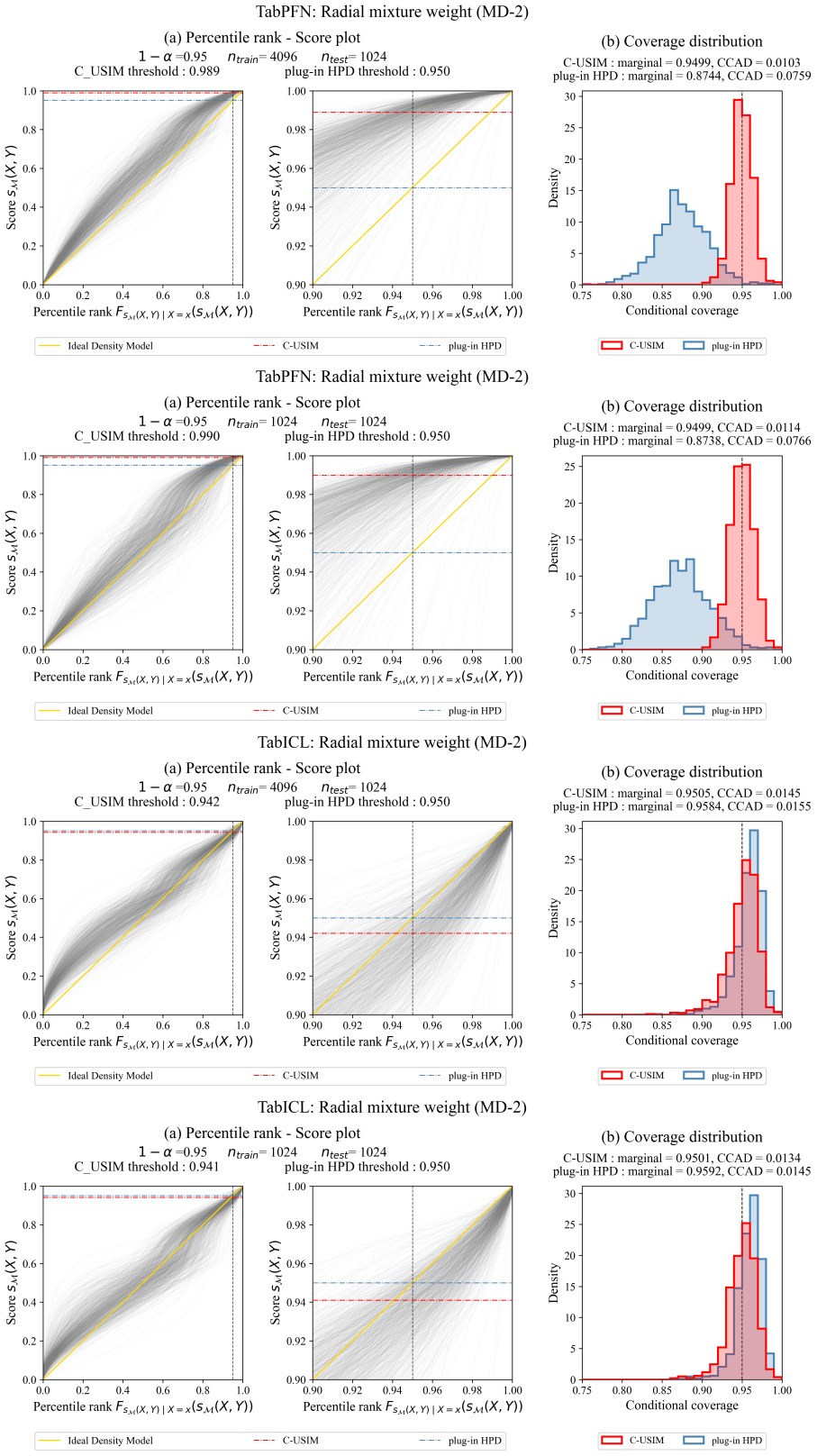

Figure 9: Percentile rank–score diagnostics for MD-2, seed 2026: C-USIM ($n_{\text{cal}} \to \infty$) and plug-in HPD.

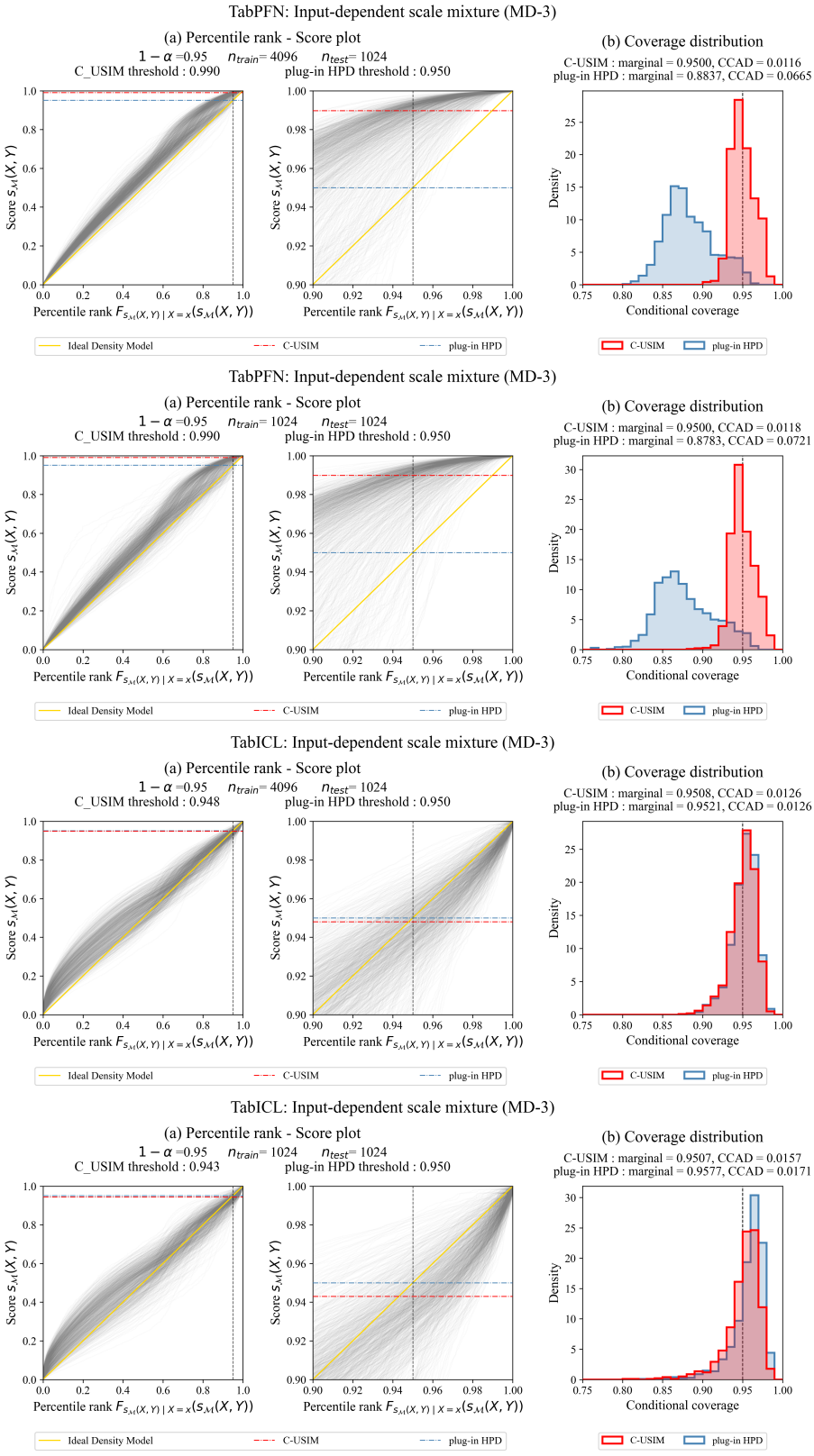

Figure 10: Percentile rank–score diagnostics for MD-3, seed 2026: C-USIM ($n_{\text{cal}} \to \infty$) and plug-in HPD.

B.4 Evaluation measure definitions

For each run, we condition on the fitted predictor, calibration data, and algorithmic randomness, treating $\widehat{C}_{1-\alpha}$ as a fixed prediction-set function. Its conditional coverage, marginal coverage, and conditional coverage absolute deviation (CCAD) are

$$c(x) := \mathbb{P}\Big(Y \in \widehat{C}_{1-\alpha}(x) \mid X = x\Big),$$
$$\mathrm{Cov} := \mathbb{E}_X[c(X)], \qquad \mathrm{CCAD} := \mathbb{E}_X[|c(X) - (1-\alpha)|].$$

The expectation is over the test covariate distribution. Below, we describe how these quantities are evaluated on synthetic and real data, followed by prediction-set length and aggregation across runs.

For synthetic data, let X_i be a test input and let $Y_{ij} \sim P(\cdot \mid X_i)$, $j = 1, \ldots, M$, be independent conditional response draws, with $M = 1000$. Both methods use the same inputs and response draws. With $\mathbf{1}\{\cdot\}$ denoting the indicator function, we compute

$$\widehat{c}_i = \frac{1}{M}\sum_{j=1}^{M} \mathbf{1}\Big\{Y_{ij} \in \widehat{C}_{1-\alpha}(X_i)\Big\},$$
$$\widehat{\mathrm{Cov}}_{\mathrm{MC}} = \frac{1}{n_{\mathrm{test}}}\sum_{i=1}^{n_{\mathrm{test}}} \widehat{c}_i, \qquad \widehat{\mathrm{CCAD}} = \frac{1}{n_{\mathrm{test}}}\sum_{i=1}^{n_{\mathrm{test}}} |\widehat{c}_i - (1-\alpha)|.$$

The empirical CCAD retains Monte Carlo error from both the test inputs and conditional response draws.

For real data, let $A_1, \ldots, A_K$ be a disjoint covariate partition learned from validation inputs alone. For the test observations (X_i, Y_i), define $I_g = \{i : X_i \in A_g\}$, $n_g = |I_g|$, and

$$h_i = \mathbf{1}\Big\{Y_i \in \widehat{C}_{1-\alpha}(X_i)\Big\}, \qquad \widehat{c}_g = \frac{1}{n_g}\sum_{i \in I_g} h_i \quad (n_g > 0).$$

Observed marginal coverage is $\widehat{\mathrm{Cov}}_{\mathrm{obs}} = n_{\mathrm{test}}^{-1}\sum_{i=1}^{n_{\mathrm{test}}} h_i$. We summarize absolute group coverage errors in two ways. Mean group error gives equal weight to each nonempty test group. CEC-X instead weights each group's absolute error by its share of the test observations:

$$\widehat{\mathrm{CEC}}_{X,1} = \sum_{g:\, n_g > 0} \frac{n_g}{n_{\mathrm{test}}} |\widehat{c}_g - (1-\alpha)|,$$

where empty test groups contribute zero. This is an absolute, not squared, coverage error over a finite partition and is not a direct estimate of pointwise CCAD.

We report prediction-set length as the sum of the lengths of all component intervals, excluding gaps between disjoint intervals. Infinite lengths remain in the averages. For each real-world dataset, coverage uses the entire test set, whereas mean length uses the first 256 inputs in the fixed test order, shared across seeds, models, and methods. The test sets contain 9,731 observations for Journal SJR and Allstate, and 5,372 for JP Anime.

In seed-repeated analyses, absolute gaps, CCAD, mean group error, and CEC-X are computed within each seed before averaging across seeds. Taking the absolute gap of the mean coverage can give a different result.

B.5 Experimental settings

Table 1 summarizes the main experimental settings. Settings specific to JP Anime and Allstate Claims Severity are given in Appendix B.6.3. In the synthetic and real-world method comparisons, the two methods share evaluation samples and the model random seed within each run, with a total budget of 1,536 labels and distinct contexts of 1,536 and 512 labels. At target coverage $1-\alpha = 0.95$, the calibrated cutoff in these comparisons is the 974th of 1,024 calibration scores. The split-ratio sweep instead varies the context and calibration sizes for C-USIM while keeping $n_{\mathrm{train}} + n_{\mathrm{cal}} = 1536$.

Table 1: Experimental settings. Synthetic and split-ratio sample counts are per mechanism and run; conditional response draws are additional Monte Carlo samples. Split-ratio coverage and CCAD use the known conditional CDF rather than the stored response draws. A dash denotes an unused setting.

Setting	Synthetic	Journal SJR	Split ratio
Dataset size	1,792 per run	27,803 cleaned rows	1,792 per run
Total label budget	1,536	1,536	1,536
Input dimensions	1, 5, 10, 20	7 (categorical)	1, 5, 10, 20
Run seeds	100–109	12100–12299	25600–25649
Number of seeds	10	200	50
Plug-in context observations	1,536	1,536	–
C-USIM context observations	512	512	256–1,280
Calibration observations	1,024	1,024	256–1,280
Validation observations	–	512	–
Test observations	256	9,731	256
Conditional draws per test input	1,000	–	1,000 (stored)
Target coverage	95%	95%	95%
Group counts (K)	–	5, 10, 15, 20, 30, 40	–
Clustering seeds	–	1717, 2717, 3717, 4717, 5717	–
Representative grouping	–	$K = 10$, seed 1717	–

Split-ratio training–calibration allocations: 256:1280, 512:1024, 768:768, 1024:512, 1229:307, and 1280:256. The fifth allocation is the closest integer split to 8:2. All six synthetic mechanisms are evaluated with both models.

Synthetic sampling. Conditional response draws are independent at each test input and shared across models and allocation arms. Table 1 reports the number of test inputs and response draws; generating equations are given in Appendix B.2.

Journal SJR preprocessing and groups. The categorical inputs are publication type, country, region, publisher, coverage years, subject categories, and subject areas. The response and prediction-set lengths use the $\log_{10}(H\text{-index} + 1)$ scale. For TFM inference, missing categories receive a dedicated token, and categorical inputs are retained. Each seed uses the same context/calibration row allocation for both models. For each fitted context, all query covariates are processed in one prediction call: 9,731 test inputs for plug-in HPD, and 1,024 calibration plus 9,731 test inputs for C-USIM. Calibration scores and the cutoff are computed from this joint output, using the finite-density reconstruction described above. For Journal SJR, the validation and test sets remain fixed across run seeds and are shared by both models. For group construction, we impute missing categorical values with their validation-set modes and apply one-hot encoding before fitting K-means with 10 initializations per clustering seed. The imputer, encoder, and cluster centers are fitted on validation covariates alone, without using responses. Test covariates undergo the same transformation and are assigned to their nearest cluster centers by Euclidean distance in the encoded feature space; categories absent from validation are encoded as all zeros in the corresponding feature block. We evaluate every combination of the six group counts and five clustering seeds in Table 1, giving 30 groupings in total. Each grouping uses its specified number of groups, with group assignments shared by both models and methods and reused across run seeds. The representative grouping was chosen before the expanded evaluation. For each seed, we compute $|\widehat{c}_g - 0.95|$ for every nonempty test group before averaging.

Numerical implementation and checks. Both models are configured with eight estimators and FP32 parameters and outputs. For one-dimensional inputs, the default TabICL ensemble generator produces two distinct preprocessing configurations; for the multidimensional inputs it produces eight. TabICL computes scaled dot-product attention in FP64 before returning FP32 activations; this numerical policy is shared by the synthetic and real-world studies. For every synthetic fitted context, numerical checks compare repeated predictions and predictions under singleton, short, extreme-mixed, and permuted queries. For every real-world seed and fitted context, we permute the entire query table, restore the original row order, and compare predictions at every query input. We also verify that each ensemble forward receives the full query table; internal batching over ensemble members preserves all query rows. For the additional datasets, FP64 attention may also be batched

over independent leading batch items to limit memory use; the query and key dimensions of each attention problem are kept intact. All checks use the fixed tolerance $10^{-4} + 10^{-4}|\text{reference}|$. These checks support the reported numerical implementation but do not establish query independence for every possible input.

Computational cost. The lightweight aspect of C-USIM is that, once the joint calibration and test outputs are available, it requires only post-processing, with no additional training or model inference. No separate density or score-correction model is fitted. For a reconstructed density with B intervals, the implementation sorts density levels once and computes cumulative probability masses, combining all intervals with equal density into the same score level. This requires $O(B \log B)$ time and $O(B)$ storage per query; extracting the prediction set from these quantities takes $O(B)$ time. Calibration adds an empirical order-statistic calculation. These costs exclude TFM inference.

B.6 Detailed experimental results

We report the synthetic results first, followed by Journal SJR and the two additional real-world datasets. The tables summarize performance across seeds, and the figures show the corresponding distributions and illustrative prediction sets.

B.6.1 Synthetic

Table 2 and Figure 17 summarize the six selected mechanisms over ten seeds per model. Figures 11–16 show prediction sets and estimated conditional-coverage distributions for each mechanism, with TabPFN above TabICL. These illustrations use seed 100, fixed before evaluation. The comparison allocates the same 1,536 labels to either the plug-in context or the C-USIM context and calibration sets; the fitted predictors therefore differ between methods.

For TabICL, marginal coverage improves on 1D-1 while CCAD increases. Undercoverage and overcoverage across inputs can offset each other in marginal coverage, whereas both contribute to the absolute errors measured by CCAD. With the training context and the resulting TFM predictive distributions held unchanged, raising only the shared calibration threshold in the rank–score plot can reduce undercoverage at some inputs while increasing overcoverage at others. If the added overcoverage outweighs the reduction in undercoverage on average, CCAD increases even as marginal coverage approaches the target. This balance depends on the model-induced score distributions across inputs. The fitted predictors also differ in the present comparison, so the observed CCAD difference does not isolate the effect of threshold adjustment.

Reading the prediction-set plots. For each model and mechanism, plots (a) and (b) show Plug-in HPD(1536) and C-USIM, respectively. They display 80 of the 256 test inputs, selected at evenly spaced ranks of the first covariate using the same rows for both methods and models. Each prediction set is centered at its observed test response, so the black vertical line at zero marks that response; gray sets miss it. The two methods share an interval-axis range within each model and mechanism. All finite interval endpoints are retained, and arrows indicate unbounded components. In the multidimensional examples, ordering by the first covariate does not define a one-dimensional conditional slice.

Reading the coverage distributions. Plot (c) uses all 256 test inputs with equal weight. Conditional coverage at each input is estimated from 1,000 shared draws from the known conditional response distribution. Both methods use the same 35 histogram bins, spanning their combined observed range with padding; no observations are removed. A dashed line marks the 0.95 target.

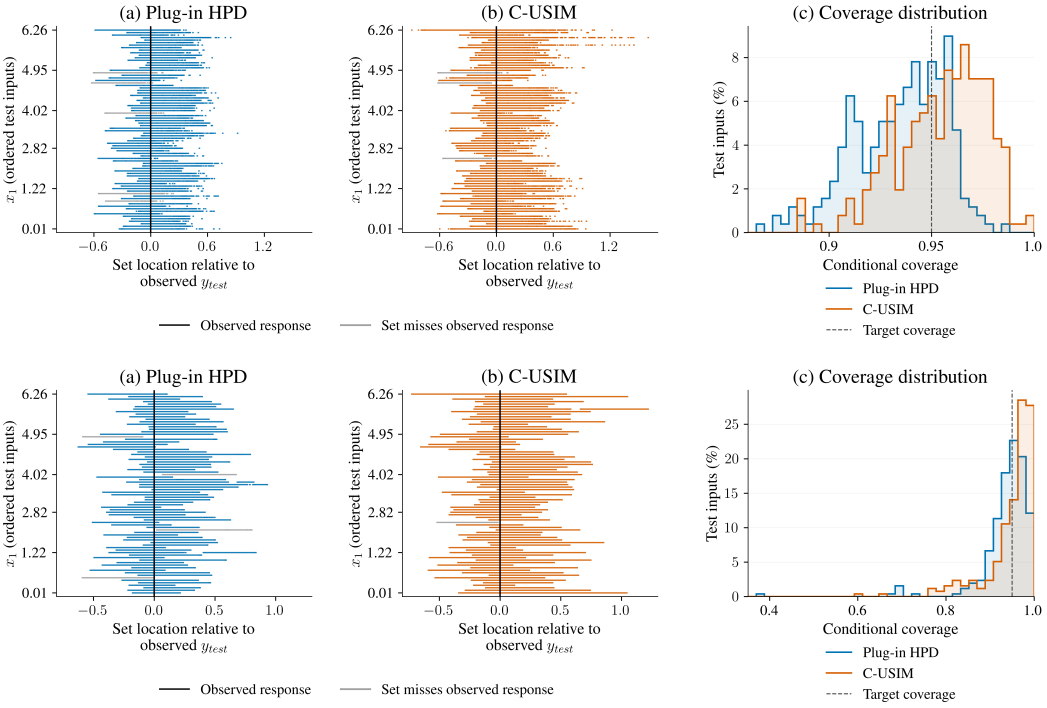

Figure 11: 1D-1, seed 100. Plotting details: Appendix B.6.1.

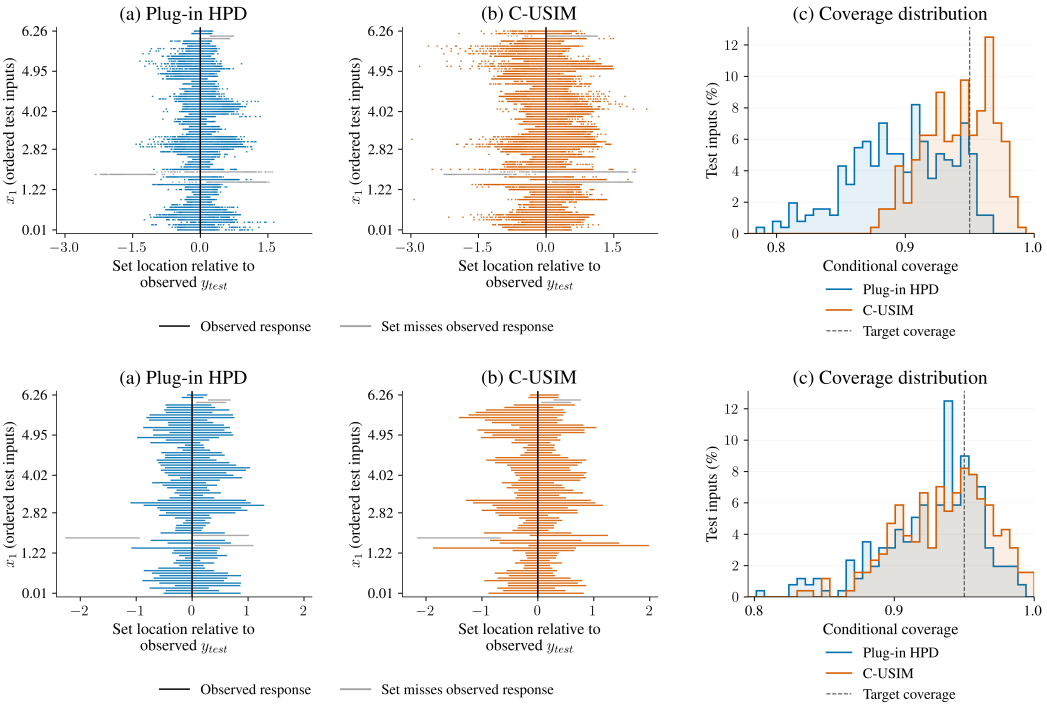

Figure 12: 1D-2, seed 100. Plotting details: Appendix B.6.1.

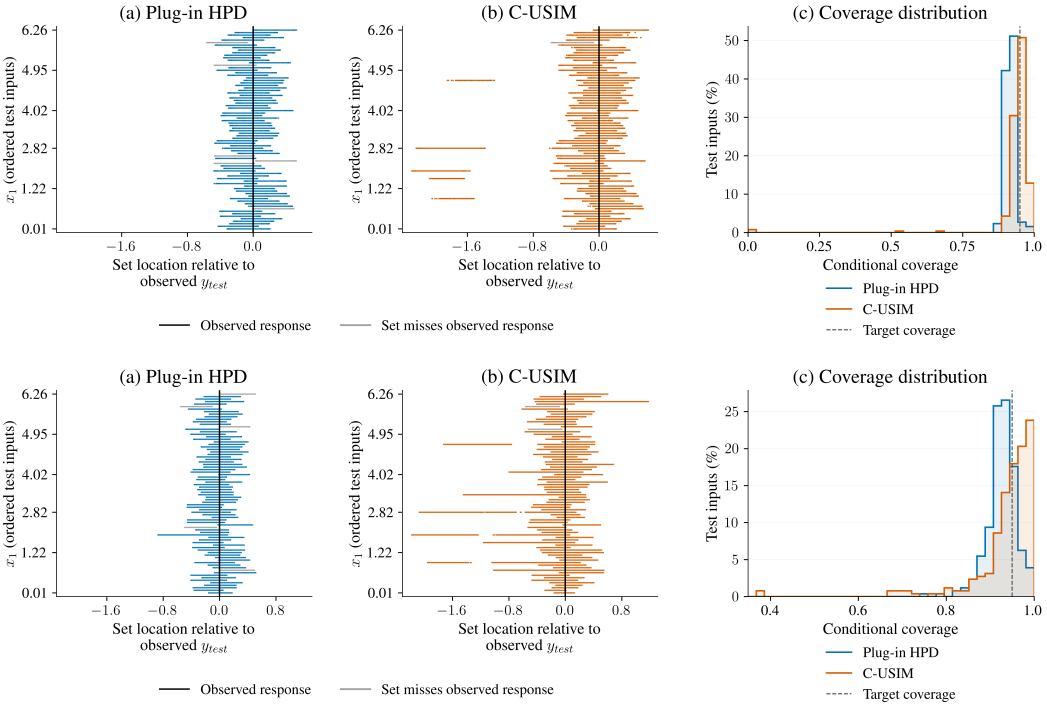

Figure 13: 1D-3, seed 100. Plotting details: Appendix B.6.1.

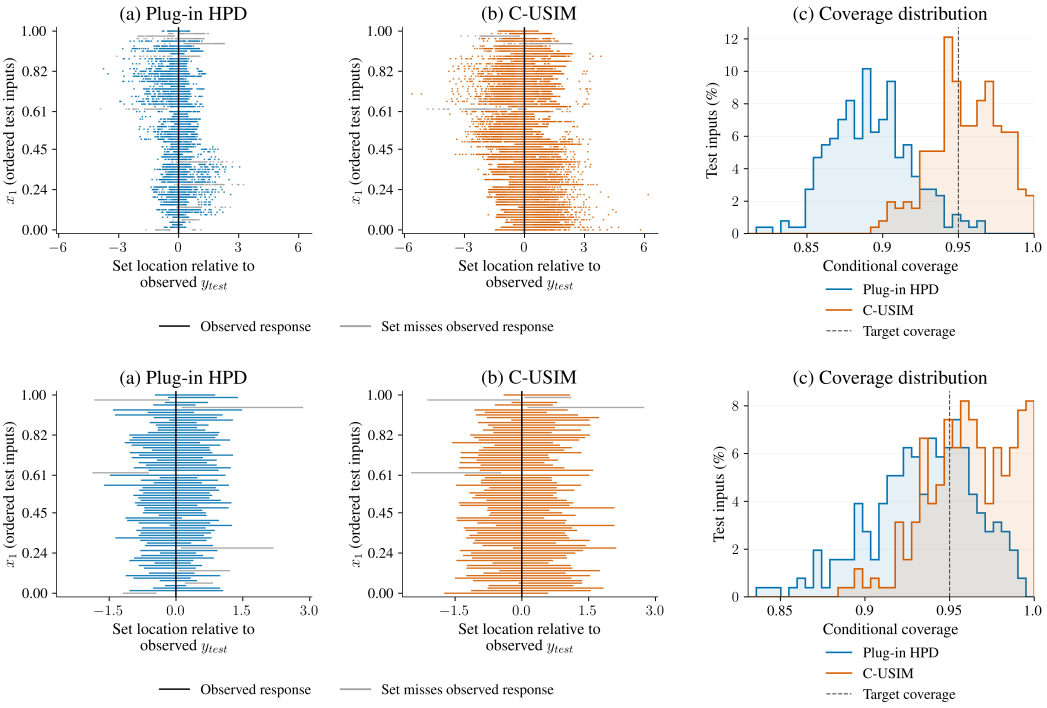

Figure 14: MD-1, seed 100. Plotting details: Appendix B.6.1.

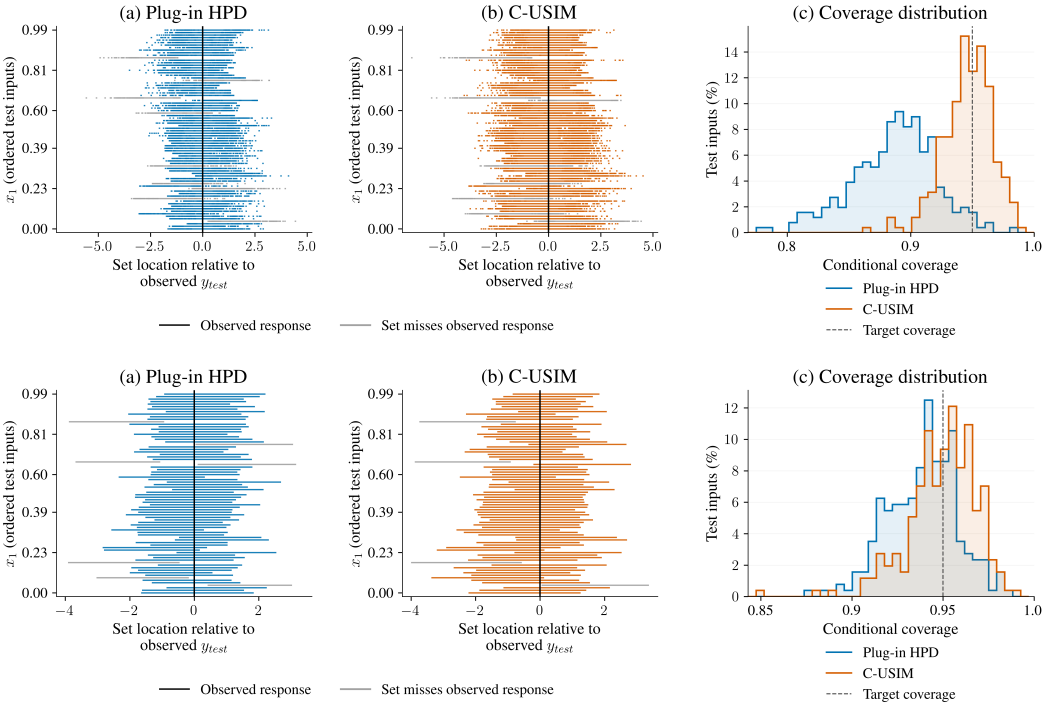

Figure 15: MD-2, seed 100. Plotting details: Appendix B.6.1.

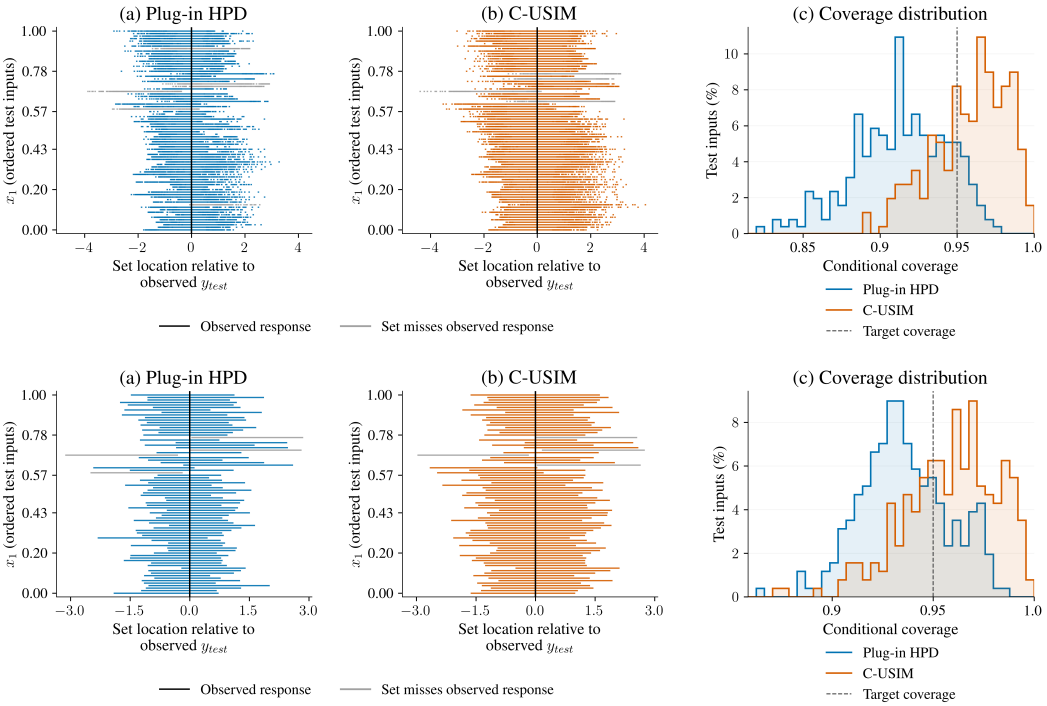

Figure 16: MD-3, seed 100. Plotting details: Appendix B.6.1.

Table 2: Synthetic comparison under a fixed budget of 1,536 labels, averaged over ten seeds per model and case. Each arrow denotes Plug-in HPD(1536) → C-USIM(512+1024). Coverage is in percent and CCAD in percentage points (pp). Cases are defined in Appendix B.2.

Model	Example	Coverage (%)	CCAD (pp)	Mean length
		Plug-in HPD(1536) → C-USIM		
TabPFN	1D-1	91.92 → 95.28	3.572 → 2.159	0.571 → 0.717
	1D-2	89.77 → 94.74	5.384 → 2.186	0.754 → 1.290
	1D-3	91.63 → 94.91	3.513 → 1.708	0.483 → 0.580
	MD-1	89.05 → 95.33	5.986 → 2.201	0.915 → 2.100
	MD-2	88.24 → 94.81	6.854 → 1.642	2.274 → 3.313
	MD-3	89.64 → 95.07	5.533 → 2.054	1.706 → 2.706
TabICL	1D-1	94.19 → 95.31	2.961 → 3.385	0.617 → 0.747
	1D-2	93.48 → 94.93	2.741 → 2.632	1.044 → 1.287
	1D-3	92.93 → 95.13	3.186 → 3.168	0.535 → 0.725
	MD-1	93.08 → 95.43	2.930 → 2.644	1.402 → 1.901
	MD-2	93.42 → 95.01	2.220 → 1.572	2.759 → 3.109
	MD-3	93.11 → 95.17	2.679 → 2.266	2.087 → 2.609

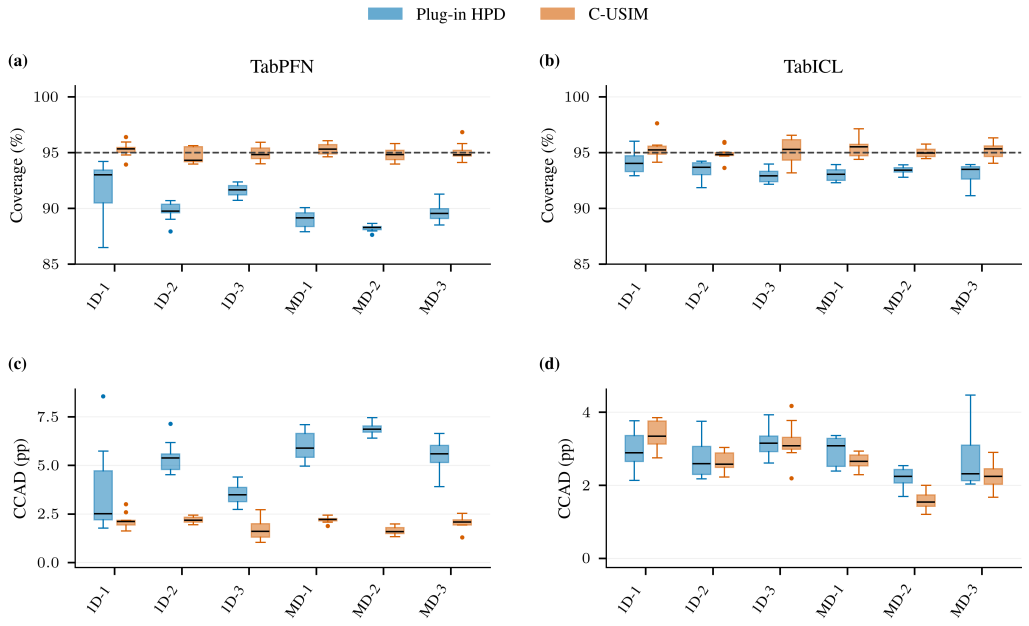

Figure 17: Synthetic coverage (a,b) and CCAD (c,d) across ten seeds for Plug-in HPD(1536) and C-USIM(512+1024); methods and label budgets follow Table 2. Boxes show Q1–Q3 and medians, with 1.5-IQR whiskers and all outliers. Dashed lines mark 95% coverage. CCAD is in percentage points on model-specific scales. These seed distributions are not confidence intervals.

B.6.2 JOURNAL SJR

Table 3 and Figure 18 report the fixed-budget comparison between Plug-in HPD(1536) and C-USIM for Journal SJR.

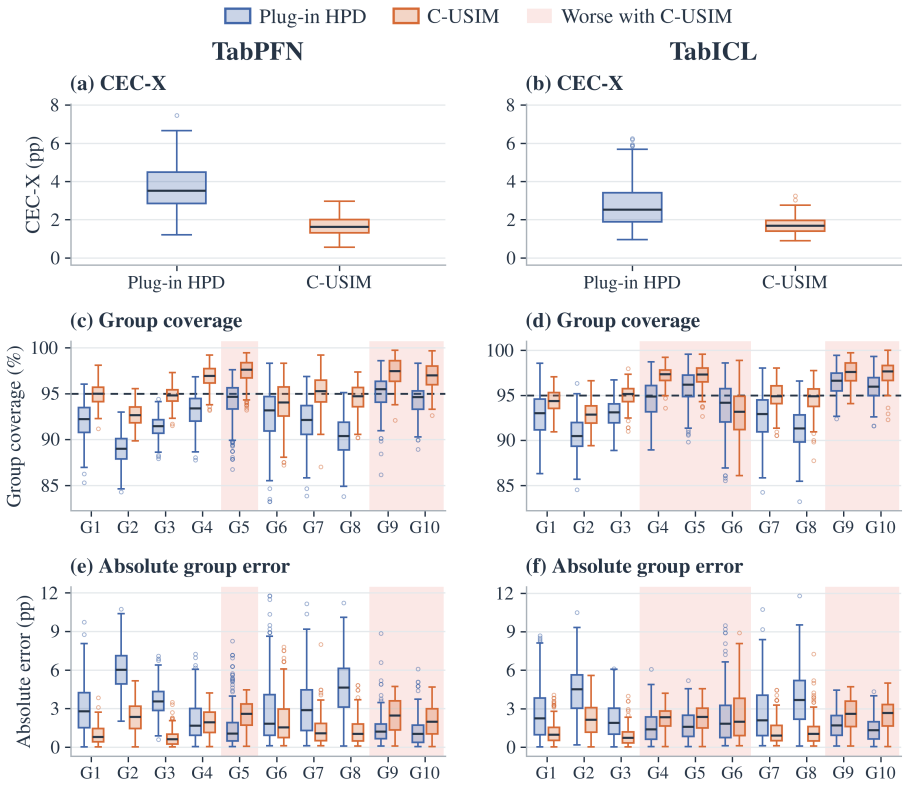

Figure 18: Journal SJR across 200 seeds for Plug-in HPD(1536) and C-USIM(512+1024); methods and label budgets follow Table 3. (a,b) CEC-X, weighted by group size; (c,d) group coverage; (e,f) absolute group error. Errors are computed within each seed. Boxes show Q1–Q3 and medians, with 1.5-IQR whiskers and all outliers, not confidence intervals. Red bands mark higher mean absolute group error with C-USIM; dashed lines mark 95% coverage. Errors are in percentage points. CEC-X shares a scale across models; group plots use model-specific scales with visual padding below zero.

Variation across groups and groupings under a fixed label budget. In the representative grouping, mean absolute error decreases in 7 of the ten groups for TabPFN and 5 for TabICL; 62.20% and 55.35% of seed–group pairs, respectively, move closer to 95% coverage. Across the 30 groupings, C-USIM reduces absolute coverage error in 47.56–70.60% of seed–group pairs for TabPFN and 40.64–61.00% for TabICL. All nonempty groups remain in the reported averages. These aggregate gains coexist with higher errors in some covariate groups and longer mean prediction sets.

B.6.3 ADDITIONAL REAL-WORLD DATASETS

Datasets and experimental settings. We add JP Anime from CARTE (Kim et al., 2024) and Allstate Claims Severity from OpenML (dataset 42571, version 1).[4] JP Anime predicts the supplied natural logarithm of the anime score from ten features: genres, type, episodes, producers, studios, source, duration, content rating, start date, and end date. Eight features are categorical; episodes and duration are numeric. The prepared cohort retains the first row for each distinct feature profile. We use the source response without applying another logarithm, and report prediction-set lengths

[4] Allstate source: `https://www.openml.org/d/42571`.

Table 3: Journal SJR under a fixed budget of 1,536 labels at 95% target coverage, averaged over 200 seeds. Each arrow denotes Plug-in HPD(1536) → C-USIM(512+1024). Absolute marginal and group gaps and CEC-X are computed within each seed before averaging, and are reported in percentage points (pp).

Measure	TabPFN	TabICL
	Plug-in HPD(1536) → C-USIM	
Marginal coverage (%)	91.50 → 94.83	92.93 → 94.87
Mean marginal gap (pp)	3.499 → 0.553	2.241 → 0.611
Mean group gap (pp)	2.979 → 1.760	2.458 → 1.856
CEC-X (pp)	3.698 → 1.662	2.782 → 1.707
Mean set length	1.149 → 1.453	1.255 → 1.540

on this log-score scale. Allstate predicts the supplied claim loss from 116 categorical and 14 continuous anonymized features. Claim identifiers are excluded from the predictors, and repeated feature profiles with distinct claim identifiers remain separate observations. No response transformation is applied to Allstate.

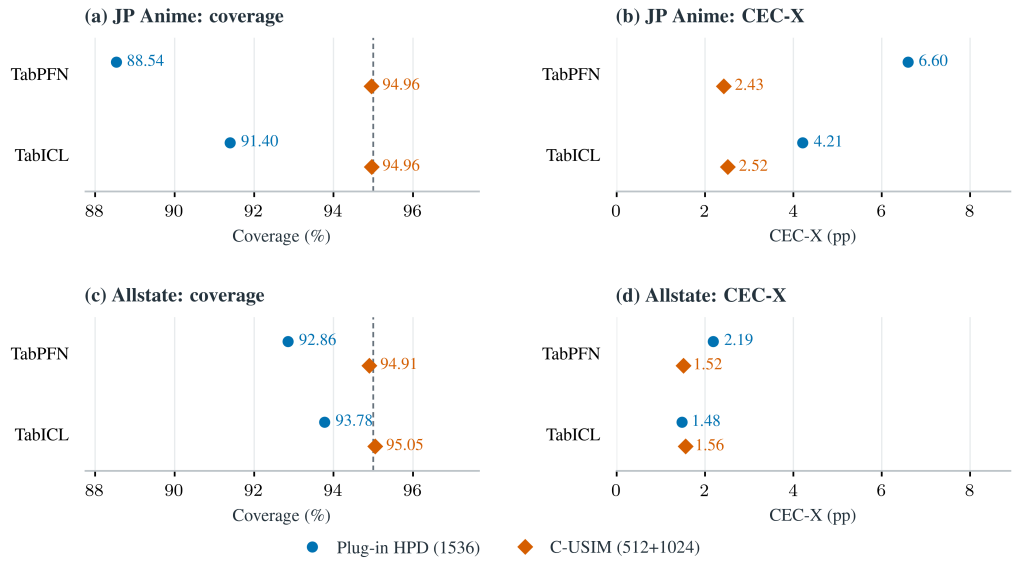

Figure 19: Additional real-world examples under a total budget of 1,536 labels. Points are means over 50 seeds, not confidence intervals. (a,b) JP Anime; (c,d) Allstate Claims Severity. Dashed lines mark 95% coverage. CEC-X uses the representative covariate grouping and is computed within each seed before averaging. Settings and detailed results are in Table 4 and Appendix B.6.3.

Table 4 gives the additional sample counts. Both datasets use 50 run seeds, 12100–12149, fixed before the new evaluation, with the same model and label-budget settings as Journal SJR. For each fitted context, plug-in HPD processes the full test set in one prediction call, and C-USIM processes all 1,024 calibration covariates together with that test set. Calibration scores are computed anew for each fitted context and joint query table. For model inference, missing numeric values are imputed using context medians and missing categorical values receive a dedicated token. Groups use the same 30 combinations of group counts and clustering seeds as Journal SJR. Numeric variables are median-imputed and standardized, while categorical variables are mode-imputed and one-hot encoded; all preprocessing and cluster centers are fitted on the 512 validation covariates only. The representative grouping remains $K = 10$ with clustering seed 1717.

Figure 19 summarizes the results, and Table 5 reports the comparison between Plug-in HPD(1536) and C-USIM(512+1024) over 50 seeds per model. For both models on both datasets, C-USIM reduces mean absolute marginal coverage error to approximately 0.54–0.60 percentage points. On JP Anime, CEC-X decreases from 6.598 to 2.429 percentage points for TabPFN and from 4.211

Table 4: Additional real-world datasets.

Dataset	Rows	Features	Categorical	Test rows	Response scale
JP Anime	15,351	10	8	5,372	$\ln(\text{Score})$
Allstate	188,318	130	116	9,731	Claim loss

to 2.521 for TabICL. On Allstate, it decreases from 2.186 to 1.517 for TabPFN. Mean prediction-set length increases in all four comparisons; lengths are measured on the dataset-specific response scales in Table 4.

Variation across groups and groupings. In the representative grouping, mean absolute group error decreases in 7 of ten groups for TabPFN and 5 for TabICL on JP Anime, and in 8 and 5 groups, respectively, on Allstate. Across all 30 groupings, CEC-X decreases for both models on JP Anime and for TabPFN on Allstate. The Allstate TabICL comparison is more sensitive to grouping: CEC-X decreases in 17 of the 30 groupings. In the representative grouping, its CEC-X changes from 1.481 to 1.561 percentage points, while mean group error changes from 1.540 to 1.523. The group plots in Figures 22 and 23 retain all ten groups and place both plug-in baselines alongside C-USIM.

Table 5: Additional real-world datasets under a fixed total budget of 1,536 labels, averaged over 50 seeds. Each arrow denotes Plug-in HPD(1536) $\rightarrow$ C-USIM(512+1024). Marginal and group errors are computed within each seed before averaging; group measures use the representative grouping. Length uses the first 256 fixed test inputs.

Dataset	Measure	TabPFN	TabICL
		Plug-in HPD (1536) $\rightarrow$ C-USIM	
JP Anime	Marginal coverage (%)	$88.54 \rightarrow 94.96$	$91.40 \rightarrow 94.96$
	Mean marginal gap (pp)	$6.461 \rightarrow 0.590$	$3.598 \rightarrow 0.540$
	Mean group gap (pp)	$6.224 \rightarrow 2.770$	$4.235 \rightarrow 2.837$
	CEC-X (pp)	$6.598 \rightarrow 2.429$	$4.211 \rightarrow 2.521$
	Mean set length	$0.447 \rightarrow 0.565$	$0.494 \rightarrow 0.539$
Allstate	Marginal coverage (%)	$92.86 \rightarrow 94.91$	$93.78 \rightarrow 95.05$
	Mean marginal gap (pp)	$2.142 \rightarrow 0.601$	$1.226 \rightarrow 0.540$
	Mean group gap (pp)	$2.240 \rightarrow 1.515$	$1.540 \rightarrow 1.523$
	CEC-X (pp)	$2.186 \rightarrow 1.517$	$1.481 \rightarrow 1.561$
	Mean set length	$5862.19 \rightarrow 6831.03$	$6195.59 \rightarrow 6883.54$

B.7 Split-ratio sensitivity

The sweep follows the synthetic protocol in Appendix B.5, with allocations and seeds listed in Table 1. Within each mechanism and seed, all allocations and both models share a labeled pool and independent test inputs: the first n_{train} pool observations form the context and the remainder form the calibration set.

Table 6: Split-ratio sensitivity across all six selected mechanisms with 1,536 total labels. Entries are means over the same 50 seeds. Gap is the per-seed absolute deviation of marginal coverage from 95%; gap and CCAD use percentage points (pp).

Case	TabPFN		TabICL	
	Gap (pp)	CCAD (pp)	Gap (pp)	CCAD (pp)
	1229:307 ($\simeq$ 8:2) $\rightarrow$ 512:1024 (1:2)			
1D-1	$1.033 \rightarrow 0.554$	$2.110 \rightarrow 2.152$	$0.911 \rightarrow 0.440$	$2.830 \rightarrow 3.381$
1D-2	$0.964 \rightarrow 0.497$	$1.998 \rightarrow 2.093$	$1.013 \rightarrow 0.561$	$2.370 \rightarrow 2.541$
1D-3	$0.857 \rightarrow 0.546$	$1.231 \rightarrow 1.379$	$1.098 \rightarrow 0.552$	$2.486 \rightarrow 3.104$
MD-1	$1.026 \rightarrow 0.583$	$2.271 \rightarrow 2.218$	$1.119 \rightarrow 0.650$	$2.569 \rightarrow 2.634$
MD-2	$0.924 \rightarrow 0.500$	$1.626 \rightarrow 1.543$	$0.969 \rightarrow 0.640$	$1.676 \rightarrow 1.606$
MD-3	$1.223 \rightarrow 0.624$	$1.784 \rightarrow 1.931$	$1.223 \rightarrow 0.562$	$2.045 \rightarrow 2.155$

Coverage and CCAD follow Appendix B.4, with conditional coverage computed using the known CDF rather than response sampling. Mean-function RMSE is $\{256^{-1} \sum_i [\hat{m}(X_i) - m(X_i)]^2\}^{1/2}$,

where $\hat{m}$ is the mean of the reconstructed predictive density and m is the true conditional mean. Metrics are computed within each seed before averaging. Table 6 reports all six mechanisms, including cases with worse CCAD.

B.8 Supplementary comparison with identical predictive densities

The main experiments compare Plug-in HPD(1536) with C-USIM(512+1024) under the same total label budget. Here, we additionally compare Plug-in HPD(512) with C-USIM to examine the effect of calibrating the cutoff while holding the reconstructed predictive densities fixed. This supplementary comparison uses different numbers of response labels: Plug-in HPD(512) uses only the 512 context labels, whereas C-USIM additionally uses 1,024 held-out calibration responses.

Settings and shared predictions. Within each run, Plug-in HPD(512) and C-USIM share the context observations, model randomness, query covariates, and reconstructed densities. The plug-in regions use the boundary density level needed to attain at least 95% model mass; C-USIM uses the 974th of the 1,024 calibration scores. Both use the finite-density reconstruction in Appendix B.1 and the evaluation measures in Appendix B.4. The data splits and seeds follow Table 1, with the additional real-world datasets specified in Table 4. Synthetic comparisons use the same 1,000 conditional response draws at each test input across all three configurations.

For Journal SJR, both 512-context configurations use the same joint table of 1,024 calibration and 9,731 test covariates in one prediction call. Plug-in HPD(512) uses the test densities from this shared output without using the calibration responses. Coverage uses all 9,731 test rows, and mean prediction-set length uses the first 256 inputs in the fixed test order for all three configurations. The inference and density reconstruction steps are shared; calibration changes the cutoff used to extract the set without fitting an additional model.

B.8.1 Synthetic

Table 7 compares Plug-in HPD(512) and C-USIM, using the same reconstructed densities and conditional response draws in each run. Mean CCAD decreases from 5.053 to 1.992 percentage points for TabPFN and from 3.540 to 2.611 for TabICL when averaged equally across the six mechanisms. Mean CCAD decreases in all twelve model–mechanism combinations, and mean prediction-set length increases in each combination. Figure 20 places this comparison alongside Plug-in HPD(1536), showing how the two context sizes affect the uncalibrated regions and how calibration changes coverage and length at the 512 context.

This comparison also clarifies the TabICL result discussed in Appendix B.6.1 on 1D-1: mean CCAD decreases from 4.313 to 3.385 percentage points after calibration, whereas Plug-in HPD(1536) has a lower CCAD of 2.961. Thus, the increase relative to the larger-context baseline cannot be attributed to threshold calibration alone.

Table 7: Synthetic comparison using identical predictive densities from a 512-observation context. Each arrow denotes Plug-in HPD(512) $\rightarrow$ C-USIM(512+1024), averaged over ten seeds. C-USIM additionally uses 1,024 calibration responses.

Model	Example	Coverage (%)	CCAD (pp)	Mean length
		Plug-in HPD(512) $\rightarrow$ C-USIM		
TabPFN	1D-1	$92.95 \rightarrow 95.28$	$2.947 \rightarrow 2.159$	$0.618 \rightarrow 0.717$
	1D-2	$90.24 \rightarrow 94.74$	$5.074 \rightarrow 2.186$	$0.805 \rightarrow 1.290$
	1D-3	$92.11 \rightarrow 94.91$	$3.268 \rightarrow 1.708$	$0.516 \rightarrow 0.580$
	MD-1	$88.49 \rightarrow 95.33$	$6.569 \rightarrow 2.201$	$0.841 \rightarrow 2.100$
	MD-2	$88.42 \rightarrow 94.81$	$6.714 \rightarrow 1.642$	$2.281 \rightarrow 3.313$
	MD-3	$89.77 \rightarrow 95.07$	$5.747 \rightarrow 2.054$	$1.783 \rightarrow 2.706$
TabICL	1D-1	$92.77 \rightarrow 95.31$	$4.313 \rightarrow 3.385$	$0.643 \rightarrow 0.747$
	1D-2	$91.98 \rightarrow 94.93$	$3.915 \rightarrow 2.632$	$0.970 \rightarrow 1.287$
	1D-3	$92.74 \rightarrow 95.13$	$4.058 \rightarrow 3.168$	$0.617 \rightarrow 0.725$
	MD-1	$92.58 \rightarrow 95.43$	$3.611 \rightarrow 2.644$	$1.343 \rightarrow 1.901$
	MD-2	$93.57 \rightarrow 95.01$	$2.222 \rightarrow 1.572$	$2.833 \rightarrow 3.109$
	MD-3	$93.41 \rightarrow 95.17$	$3.123 \rightarrow 2.266$	$2.278 \rightarrow 2.609$

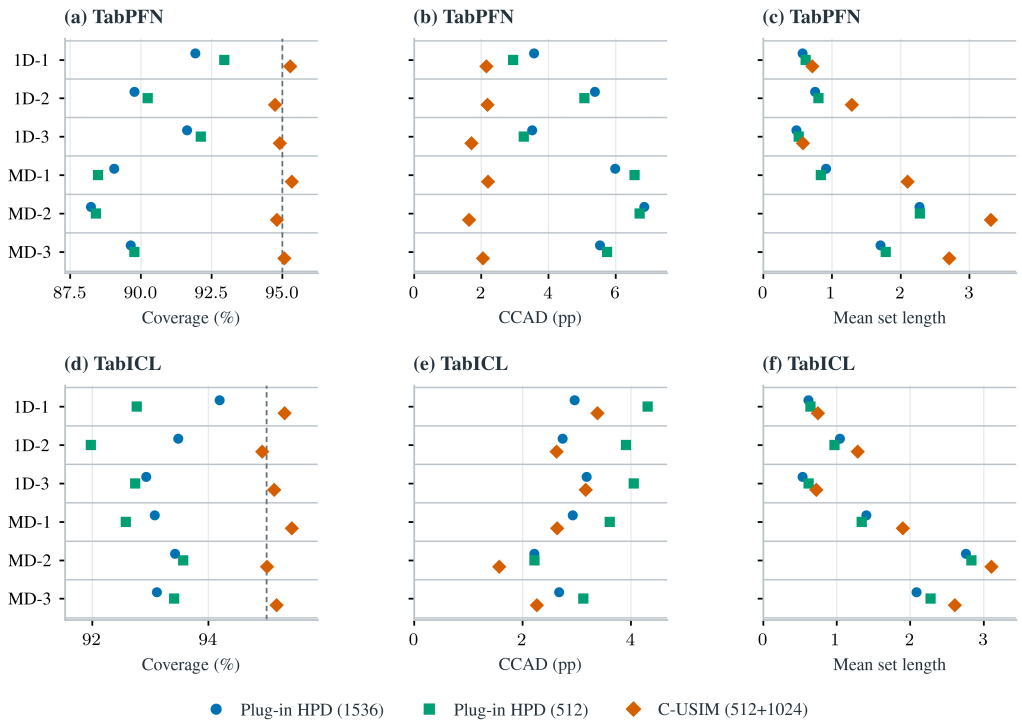

Figure 20: Synthetic coverage, CCAD, and prediction-set length for all three configurations. Points are means over ten seeds, not confidence intervals. Horizontal lines separate mechanisms. The two 512-context methods share predictive densities. Dashed lines mark 95% coverage. All interval components contribute to length, excluding gaps.

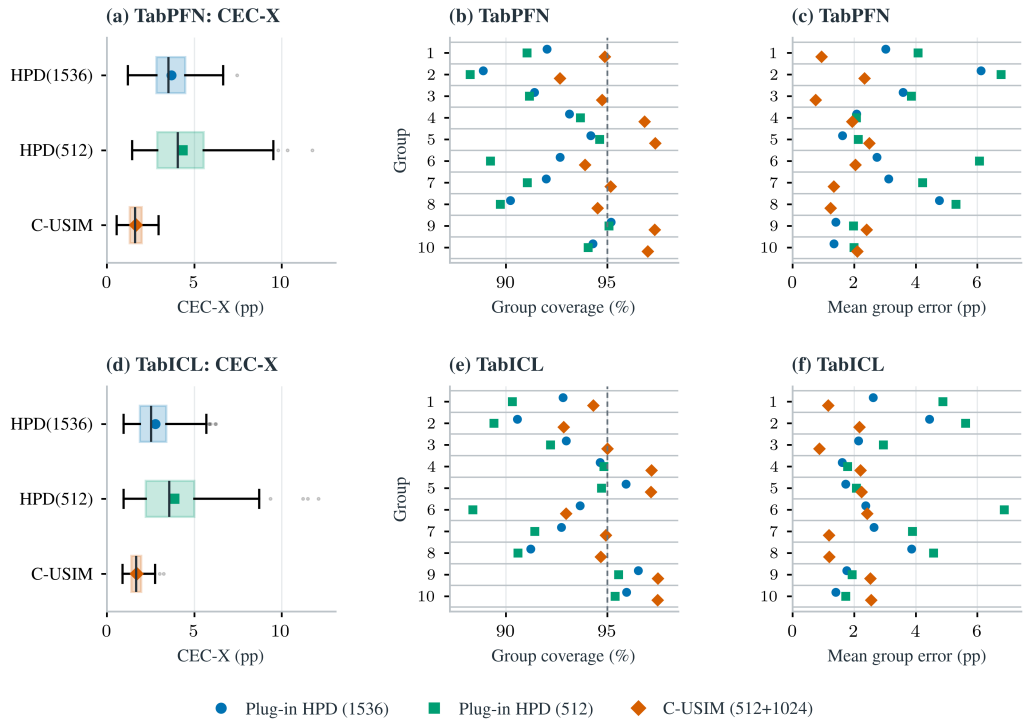

Figure 21: Journal SJR with both plug-in baselines and C-USIM, using the representative grouping. (a,d) CEC-X over 200 seeds: boxes show quartiles and medians, whiskers extend to 1.5 IQR, all outliers are retained, and colored markers denote means. (b,e) Mean coverage in each group. (c,f) Mean within-seed absolute group error. Horizontal lines in the group plots separate groups. These summaries are not confidence intervals. Dashed lines mark 95% coverage; all ten groups are retained.

Table 8: Journal SJR using identical predictive densities from a 512-observation context, averaged over 200 seeds. Each arrow denotes Plug-in HPD(512) → C-USIM(512+1024). Both configurations use the same joint calibration/test query table; only C-USIM uses the 1,024 calibration responses. Group errors use the representative grouping. Coverage uses all 9,731 test rows; length uses the first 256 inputs in the fixed test order.

Measure	TabPFN	TabICL
	Plug-in HPD(512) → C-USIM	
Marginal coverage (%)	91.06 → 94.83	91.72 → 94.87
Mean marginal gap (pp)	3.964 → 0.553	3.431 → 0.611
Mean group gap (pp)	3.846 → 1.760	3.630 → 1.856
CEC-X (pp)	4.333 → 1.662	3.863 → 1.707
Mean set length	1.261 → 1.453	1.359 → 1.540

B.8.2 JOURNAL SJR

Table 8 reports the results over 200 seeds per model. In the representative grouping, mean CEC-X decreases from 4.333 to 1.662 percentage points for TabPFN and from 3.863 to 1.707 for TabICL. Relative to Plug-in HPD(512), C-USIM lowers mean CEC-X in all 30 groupings for each model. In the representative grouping, mean absolute group error decreases from 3.846 to 1.760 percentage points for TabPFN and from 3.630 to 1.856 for TabICL. Mean error decreases in 7 of the ten groups for TabPFN and 6 for TabICL; 66.05% and 61.50% of seed–group pairs, respectively, move closer to the target. Across all groupings, these fractions range from 46.38% to 70.00% for TabPFN and from 43.26% to 64.00% for TabICL. All nonempty groups remain in the averages and Figure 21 shows all ten representative groups.

The discussion in Appendix B.6.2 also applies here. Mean prediction-set length increases from 1.261 to 1.453 for TabPFN and from 1.359 to 1.540 for TabICL, as reported in Table 8.

B.8.3 Additional real-world datasets

Table 9 compares Plug-in HPD(512) with C-USIM using identical predictive densities. Both methods use the same joint calibration/test table for each seed; only C-USIM uses the 1,024 calibration responses. On JP Anime, mean CEC-X decreases from 4.681 to 2.429 percentage points for TabPFN and from 3.103 to 2.521 for TabICL. On Allstate, the corresponding changes are 2.667 to 1.517 and 2.277 to 1.561. CEC-X decreases in all 30 groupings for each of the four dataset–model pairs, and mean prediction-set length increases in each pair.

The Allstate TabICL result illustrates the distinction between the two comparisons: calibrating the same 512-context densities lowers CEC-X, while comparison with Plug-in HPD(1536) also changes the predictor and gives the smaller, grouping-sensitive differences described in Appendix B.6.3.

Table 9: Additional real-world datasets using identical predictive densities from a 512-observation context, averaged over 50 seeds. Each arrow denotes Plug-in HPD(512) $\rightarrow$ C-USIM(512+1024). Only C-USIM uses the additional 1,024 calibration responses. Group measures use the representative grouping; response scales and test sizes follow Table 4.

Dataset	Measure	TabPFN	TabICL
		Plug-in HPD (512) $\rightarrow$ C-USIM	
JP Anime	Marginal coverage (%)	$91.05 \rightarrow 94.96$	$94.01 \rightarrow 94.96$
	Mean marginal gap (pp)	$3.974 \rightarrow 0.590$	$1.853 \rightarrow 0.540$
	Mean group gap (pp)	$4.697 \rightarrow 2.770$	$3.330 \rightarrow 2.837$
	CEC-X (pp)	$4.681 \rightarrow 2.429$	$3.103 \rightarrow 2.521$
	Mean set length	$0.486 \rightarrow 0.565$	$0.522 \rightarrow 0.539$
Allstate	Marginal coverage (%)	$92.43 \rightarrow 94.91$	$93.03 \rightarrow 95.05$
	Mean marginal gap (pp)	$2.568 \rightarrow 0.601$	$2.001 \rightarrow 0.540$
	Mean group gap (pp)	$2.646 \rightarrow 1.515$	$2.279 \rightarrow 1.523$
	CEC-X (pp)	$2.667 \rightarrow 1.517$	$2.277 \rightarrow 1.561$
	Mean set length	$5935.33 \rightarrow 6831.03$	$6145.43 \rightarrow 6883.54$

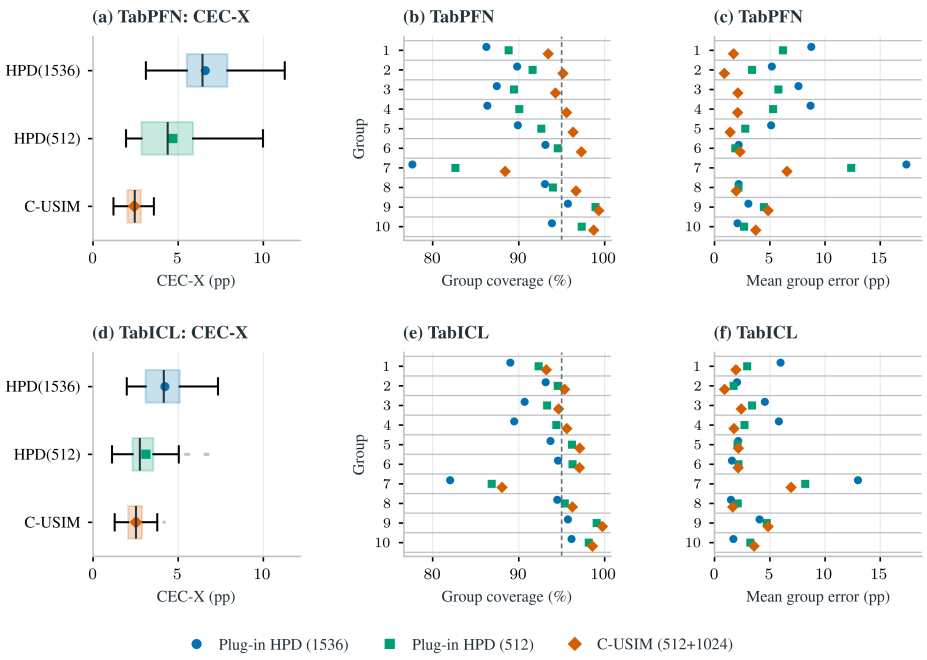

Figure 22: JP Anime with both plug-in baselines and C-USIM, using the representative grouping. (a,d) CEC-X over 50 seeds: boxes show quartiles and medians, whiskers extend to 1.5 IQR, all outliers are retained, and colored markers denote means. (b,e) Mean group coverage. (c,f) Mean within-seed absolute group error. Horizontal lines separate groups; group 9 has 15 test observations. Dashed lines mark 95% coverage. These summaries are not confidence intervals.

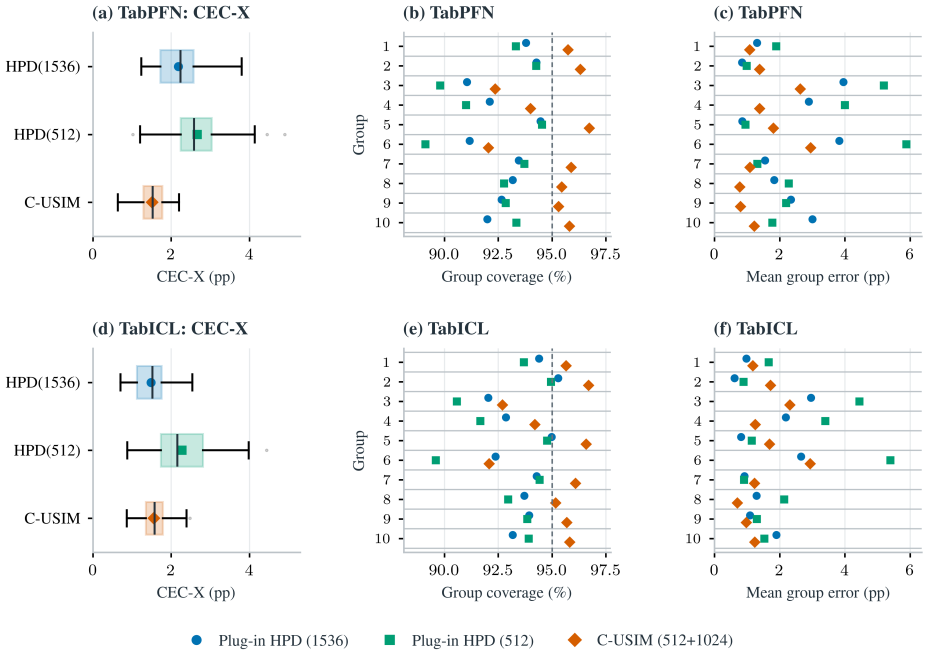

Figure 23: Allstate Claims Severity with both plug-in baselines and C-USIM, using the representative grouping. (a,d) CEC-X over 50 seeds: boxes show quartiles and medians, whiskers extend to 1.5 IQR, all outliers are retained, and colored markers denote means. (b,e) Mean group coverage. (c,f) Mean within-seed absolute group error. Horizontal lines separate all ten groups; dashed lines mark 95% coverage. These summaries are not confidence intervals.