Research Square

Preprints are preliminary reports that have not undergone peer review. They should not be considered conclusive, used to inform clinical practice, or referenced by the media as validated information.

CoCoNet: Convex Conformal Network for Multi-Dimensional Uncertainty Regions

Alireza Moradi
Georgia Institute of Technology
Thomas Bruys
Georgia Institute of Technology
Pascal Van Hentenryck
Georgia Institute of Technology
Reza Zandehshahvar
mohammadreza3@gatech.edu

Georgia Institute of Technology https://orcid.org/0000-0002-6249-905X

Research Article

Keywords: Conformal Prediction, Uncertainty Quantification, Multivariate Prediction Regions, Learned Conformity Scores, Input-Convex Neural Networks, Adaptive Prediction Sets

Posted Date: October 21st, 2026

DOI: https://doi.org/10.21203/rs.3.rs-11421328/v2

License: This work is licensed under a Creative Commons Attribution 4.0 International License. Read Full License

Additional Declarations: The authors declare no competing interests.

CoCoNet: CONVEX CONFORMAL NETWORK FOR MULTI-DIMENSIONAL UNCERTAINTY REGIONS

Alireza Moradi[1,3]*, **Thomas Bruys**[2,3], **Pascal Van Hentenryck**[1,3], **Reza Zandehshahvar**[1,3]
[1]H. Milton Stewart School of Industrial and Systems Engineering (ISyE), Georgia Institute of Technology
[2]School of Electrical and Computer Engineering (ECE), Georgia Institute of Technology
[3]NSF AI Institute for Advances in Optimization (AI4OPT), Georgia Institute of Technology
Atlanta, GA, USA
alirezamoradi@gatech.edu, thomas.bruys@gatech.edu
pascal.vanhentenryck@isye.gatech.edu, reza@isye.gatech.edu

ABSTRACT

Conformal prediction (CP) provides distribution-free coverage guarantees under exchangeability, but the efficiency of its prediction regions depends critically on the choice of the conformity score. This is particularly important in multivariate regression, where dependencies across output dimensions induce complex joint uncertainty structures. This paper introduces *CoCoNet* (***Co***nvex ***Co***nformal ***Net***work) for learning convex conformity scores that induce flexible, data-adaptive prediction regions. *CoCoNet* learns the score from held-out data by optimizing prediction-region efficiency while enforcing convexity with respect to the residual. *CoCoNet* then applies split-conformal calibration to retain finite-sample coverage guarantees. The paper presents two variants of the method: *CoCoNet*$_{\text{Spline}}$, based on piecewise-linear scores, and *CoCoNet*$_{\text{ICNN}}$, based on Input Convex Neural Networks. Experiments compare *CoCoNet* with other CP baselines over synthetic and real-world datasets with output dimensions ranging from 2 to 16. The results demonstrate the flexibility of *CoCoNet* in learning different response types. Across 8 models and 30 multi-dimensional regression tasks, *CoCoNet*$_{\text{ICNN}}$ and *CoCoNet*$_{\text{Spline-}\ell_1}$ achieve the best efficiencies with average ranks of 1.45 and 2.75, respectively. In particular, *CoCoNet*$_{\text{ICNN}}$ ranks first on 22 tasks out of 30 in terms of efficiency.

***K*eywords** Conformal Prediction, Uncertainty Quantification, Multivariate Prediction Regions, Learned Conformity Scores, Input-Convex Neural Networks, Adaptive Prediction Sets, Distribution-Free Prediction

1 Introduction

Prediction regions complement point predictions by characterizing which outcomes remain plausible for a given input, providing a principled representation of predictive uncertainty for reliable decision-making and optimization. This is particularly important in multivariate regression, where multiple, often strongly dependent, outcomes jointly influence downstream decisions. Conformal prediction (CP) provides a distribution-free and model-agnostic framework for constructing prediction regions with finite-sample marginal coverage guarantees under exchangeability assumption (Vovk et al., 2005; Lei et al., 2018).

While CP provides marginal coverage validity, the efficiency (i.e., size) and geometry of the resulting prediction regions depend critically on the choice of *conformity score* (Lei et al., 2018; Romano et al., 2019; Bai et al., 2022). Different conformity scores can achieve the same target coverage while producing regions with

*Corresponding author

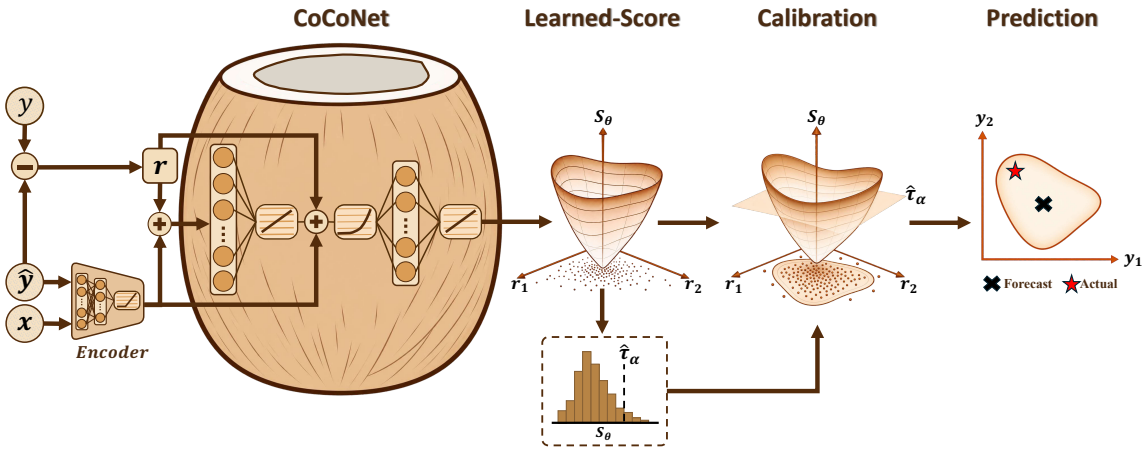

Figure 1: **Overview of *CoCoNet*.** Given covariates $\mathbf{x}$, point predictions $\widehat{\mathbf{y}}$, and actual responses $\mathbf{y}$, residuals $\mathbf{r} = \mathbf{y} - \widehat{\mathbf{y}}$ are constructed while $h(\cdot)$ encodes $\mathbf{x}$ and $\widehat{\mathbf{y}}$ into an input-dependent representation. The residuals and encoded representation are then passed through the scoring network, where they determine a conformity score S_θ that is convex in the residual space. A held-out calibration set determines the conformal threshold $\widehat{\tau}_\alpha$. The corresponding $\widehat{\tau}_\alpha$-sublevel set defines a convex region in residual space, which is translated by the point prediction $\widehat{\mathbf{y}}$ to obtain the final prediction region in the response space with finite-sample marginal coverage under exchangeability.

substantially different sizes and shapes. This distinction becomes particularly important in multivariate settings, where dependencies across output dimensions can induce complex joint uncertainty structures. A conformity score that has poor alignment with the uncertainty structure of the output can therefore produce unnecessarily large prediction regions despite satisfying the desired coverage. Designing conformity scores that capture the geometry of the joint uncertainty is thus a central challenge in constructing efficient multivariate prediction regions.

This work focuses on constructing *convex* CP regions for multi-dimensional regression. Beyond providing a coherent and interpretable representation of joint uncertainty, convexity serves as a useful form of geometric regularization: it prevents fragmented prediction regions while retaining the flexibility to adapt to different uncertainty geometries. Convex regions are also particularly desirable when uncertainty estimates inform downstream decisions. They support efficient computational operations, such as membership testing and projection, and can be directly incorporated as uncertainty sets into well-developed frameworks for robust optimization, planning, and control, substantially improving the tractability of the resulting decision-making problems (Bertsimas et al., 2011; Johnstone and Cox, 2021; Tumu et al., 2024).

Several conformal methods have been developed to construct structured convex prediction regions for multivariate regression. Coordinatewise constructions naturally produce axis-aligned hyperrectangles (Sampson and Chan, 2024; Schlembach et al., 2025), while covariance- and Mahalanobis-based scores yield ellipsoidal regions that account for dependencies among outputs (Johnstone and Cox, 2021; Messoudi et al., 2022). Minimum-volume conformal sets (MVCS) formulate conformity-score construction as an optimization problem over flexible norm-based families, including single- and multi-norm constructions (Braun et al., 2026). These approaches provide increasingly adaptive convex uncertainty regions, but their flexibility remains tied to a prespecified geometric family: the prediction region is constrained to remain a hyperrectangle, ellipsoid, norm-based set, or another predefined shape. Consequently, even when the parametric scores are learned from data, their induced geometry can adapt only within the selected family and may result in unnecessarily large prediction regions. Extended literature review is presented in Appendix A.

This paper introduces *CoCoNet*, the **Co**nvex **Co**nformal **Net**work, which learns input-adaptive convex conformity scores directly from data to construct efficient prediction regions given outputs of a base predictive model. By enforcing convexity of the learned score with respect to the residual, its sublevel sets induce convex prediction regions while allowing their geometry to adapt to the joint uncertainty structure. The score is

learned on a dataset disjoint from the conformal calibration set and is subsequently calibrated using split CP, preserving finite-sample marginal coverage under exchangeability. Figure 1 illustrates the complete pipeline, from feature encoding and conformity-score learning through calibration to construction of the final prediction region. The main contributions of this paper are:

- **Learning input-adaptive convex prediction regions:** This paper formulates the construction of convex prediction regions as a score-learning problem. It introduces a class of *admissible* conformity scores $\mathcal{S}_{\mathrm{adm}}$ whose sublevel sets are convex and contain the point prediction. It also establishes that any compact convex region containing the point prediction in its interior can be represented as a sublevel set of an admissible score. *CoCoNet* then learns an input-dependent score from held-out data using an efficiency-driven objective. This enables the geometry to be learned from data without imposing restrictions beyond convexity.
- **Flexible convex score parameterizations:** Two variants of *CoCoNet* are developed: *CoCoNet*$_{\mathrm{Spline}}$, which learns piecewise-linear convex scores, and *CoCoNet*$_{\mathrm{ICNN}}$, which parameterizes the score using an Input Convex Neural Network (ICNN) Amos et al. (2017). *CoCoNet*$_{\mathrm{Spline}}$ induces piecewise linear prediction regions including hyperrectangles (using ℓ_∞ norm aggregation across the output coordinates), or polytopes (using ℓ_1-norm aggregation). *CoCoNet*$_{\mathrm{ICNN}}$ adapts to general convex prediction regions. These variants provide different levels of flexibility for learning convex prediction-region geometries directly from data.
- **Extensive empirical evaluation:** *CoCoNet* is evaluated against existing CP baselines on synthetic instances and 30 real-world multivariate regression tasks with output dimensions ranging from 2 to 16, covering diverse dependency structures and prediction-region geometries.

Comparing with the other baselines, *CoCoNet*$_{\mathrm{ICNN}}$ achieves the best overall efficiency, ranking first on 22 of the 30 real-world multi-dimensional regression tasks and an average rank of 1.45 (out of 8 considered CP methods) while maintaining valid coverage. *CoCoNet*$_{\mathrm{Spline}\text{-}\ell_1}$ ranks as the second best approach with the average rank of 2.75 (out of 8 considered CP methods). The codes are available on GitHub [2].

2 Problem Setup

Let $\mathbf{X} \in \mathcal{X} \subseteq \mathbb{R}^d$ denote the input vector and $\mathbf{Y} \in \mathcal{Y} \subseteq \mathbb{R}^p$ a p-dimensional continuous response, with $(\mathbf{X}, \mathbf{Y})$ drawn from an unknown joint distribution $P_{\mathbf{XY}}$. A pretrained point predictor $F : \mathcal{X} \to \mathcal{Y}$ produces $\widehat{\mathbf{y}} = F(\mathbf{x})$, with residual $\mathbf{r} = \mathbf{y} - F(\mathbf{x}) \in \mathbb{R}^p$. Given the fixed predictor F, the goal is to construct a prediction region $\mathcal{C}(\mathbf{x}) \subseteq \mathcal{Y}$ from its outputs such that:

$$\mathbb{P}(\mathbf{Y} \in \mathcal{C}(\mathbf{X})) \geq 1 - \alpha, \tag{1}$$

while keeping its expected size $\mathbb{E}[|\mathcal{C}(\mathbf{X})|]$ as small as possible.

Split Conformal Prediction. Split CP constructs prediction regions with finite-sample marginal coverage under exchangeability (Vovk et al., 2005; Papadopoulos et al., 2002; Lei et al., 2018). To determine which candidate responses should be included, it uses a conformity score $S : \mathcal{X} \times \mathbb{R}^p \to \mathbb{R}$ that measures the compatibility of a candidate output with the prediction, with smaller values indicating greater conformity. A simple example is $S(\mathbf{x}, \mathbf{r}) = \|\mathbf{r}\|_1$. For a threshold τ, the score induces the prediction region

$$\mathcal{C}(\mathbf{x}; \tau) = \{\mathbf{y} \in \mathcal{Y} : S(\mathbf{x}, \mathbf{y} - F(\mathbf{x})) \leq \tau\}. \tag{2}$$

Given an independent calibration set $\mathcal{D}_{\mathrm{cal}} = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^{n_{\mathrm{cal}}}$, define

$$s_i = S(\mathbf{x}_i, \mathbf{y}_i - F(\mathbf{x}_i)), \qquad \widehat{\tau}_\alpha = s_{(k)}, \qquad k = \lceil (n_{\mathrm{cal}} + 1)(1 - \alpha) \rceil, \tag{3}$$

where $s_{(1)} \leq \cdots \leq s_{(n_{\mathrm{cal}})}$ are the ordered calibration scores. The calibrated conformal region is therefore $\mathcal{C}(\mathbf{x}; \widehat{\tau}_\alpha)$.

Proposition 1 (Marginal coverage)**.** *Let F be a fixed predictor and S a fixed conformity score, both independent of the calibration set $\mathcal{D}_{\mathrm{cal}}$. Assume that the calibration examples and the test example are exchangeable, and the scores are almost surely distinct. Then split CP constructs the prediction region $\mathcal{C}(\mathbf{X}; \widehat{\tau}_\alpha)$ such that:*

$$1 - \alpha \leq \mathbb{P}(\mathbf{Y} \in \mathcal{C}(\mathbf{X}; \widehat{\tau}_\alpha)) \leq 1 - \alpha + \frac{1}{n_{\mathrm{cal}} + 1}. \tag{4}$$

[2] `https://github.com/AI4OPT/CoCoNet`

Learning Conformity Scores. While conformal calibration guarantees distribution-free marginal coverage for any valid conformity score, the efficiency and geometric adaptivity of the calibrated prediction region depend fundamentally on the design of the score function. Instead of relying on a fixed score function S, the objective of this paper is to learn an input-dependent score function $S_{\boldsymbol{\theta}} : \mathcal{X} \times \mathbb{R}^p \to \mathbb{R}_+$ (with $\boldsymbol{\theta}$ representing the learnable parameters) which induces a prediction region $C_{\boldsymbol{\theta}}(\mathbf{X}; \tau)$ such that:

$$
\begin{aligned}
(\boldsymbol{\theta}^\star, \tau^\star) \in \underset{\boldsymbol{\theta}\in\boldsymbol{\Theta},\ \tau\geq 0}{\arg\min} \quad & \mathbb{E}\left[|\mathcal{C}_{\boldsymbol{\theta}}(\mathbf{X};\tau)|\right] \\
\text{s.t.} \quad & \mathbb{P}(\mathbf{Y} \in \mathcal{C}_{\boldsymbol{\theta}}(\mathbf{X};\tau)) \geq 1-\alpha, \\
& \mathcal{C}_{\boldsymbol{\theta}}(\mathbf{x};\tau) \text{ is convex}, \qquad \forall\, \mathbf{x} \in \mathcal{X}, \\
& F(\mathbf{x}) \in \mathcal{C}_{\boldsymbol{\theta}}(\mathbf{x};\tau), \qquad \forall\, \mathbf{x} \in \mathcal{X}.
\end{aligned} \tag{5}
$$

This formulation captures the desired objective of learning efficient prediction regions while enforcing valid coverage, convexity, and inclusion of the point prediction of a fixed predictor F. However, it cannot be optimized directly because the underlying data distribution is unknown. *CoCoNet* addresses this problem by enforcing the structural requirements through the score parameterization and learning its parameters $\boldsymbol{\theta}$ from data using an empirical surrogate objective. Once the score is learned, the final coverage guarantee is obtained through split CP calibration.

3 Methodology

3.1 Preliminaries

To address the optimization problem in (5), *CoCoNet* learns a conformity score function whose sublevel sets define efficient prediction regions. Rather than directly optimizing the geometry of these regions, the proposed framework imposes structural conditions on the score function to guarantee the desired geometric properties.

Admissible Score Functions. This paper defines a class of functions that induce prediction regions that satisfy the structural constraints in (5):

$$
\mathcal{S}_{\mathrm{adm}} = \Big\{ S_{\boldsymbol{\theta}} : \mathcal{X} \times \mathbb{R}^p \to [0,\infty) : S_{\boldsymbol{\theta}}(\mathbf{x},\cdot) \text{ is convex},\ S_{\boldsymbol{\theta}}(\mathbf{x},\mathbf{0}) = 0,\ \forall\, \mathbf{x} \in \mathcal{X} \Big\}. \tag{6}
$$

The admissible class $\mathcal{S}_{\mathrm{adm}}$ imposes only structural conditions on the conformity score: convexity, nonnegativity, and $S_\theta(x, 0) = 0$. These conditions guarantee convex prediction regions containing $F(\mathbf{x})$ while leaving the boundary geometry unrestricted within the admissible convex class.

Proposition 2 (Admissible scores induce convex prediction regions)**.** *Let $S_\theta \in \mathcal{S}_{\mathrm{adm}}$. Then, for any $\mathbf{x} \in \mathcal{X}$ and $\tau \geq 0$, the prediction region*

$$
\mathcal{C}_\theta(\mathbf{x}; S; \tau) = \{\mathbf{y} \in \mathcal{Y} : S_\theta(\mathbf{x}, \mathbf{y} - F(\mathbf{x})) \leq \tau\} \tag{7}
$$

is convex and contains the point prediction $F(\mathbf{x})$. See Appendix B.2 for proofs.

Proposition 2 converts the geometric constraints of the region optimization problem into structural conditions on the score. Proposition 3 establishes the converse: within the unrestricted admissible score class, any compact, full-dimensional convex region containing the point prediction in its interior can be represented as a score sublevel set. Proofs are provided in Appendix B.2.

Proposition 3 (Convex-region representation)**.** *Consider a given input $\mathbf{x} \in \mathcal{X}$ and let $\mathcal{C}(\mathbf{x}) \subset \mathbb{R}^p$ be a compact, full-dimensional convex region satisfying $F(\mathbf{x}) \in \operatorname{int} \mathcal{C}(\mathbf{x})$.*

There exists a nonnegative convex score $S_{\mathcal{C}}(\mathbf{x}, \cdot)$ satisfying $S_{\mathcal{C}}(\mathbf{x}, \mathbf{0}) = 0$ such that

$$
\mathcal{C}(\mathbf{x}) = \{\mathbf{y} \in \mathbb{R}^p : S_{\mathcal{C}}(\mathbf{x}, \mathbf{y} - F(\mathbf{x})) \leq 1\}. \tag{8}
$$

In particular, such a score is given by the Minkowski gauge of the translated region $\mathcal{C}(\mathbf{x}) - F(\mathbf{x})$.

Together, Propositions 2 and 3 provide the geometric foundation of *CoCoNet*: convexity is enforced through the score, while the particular convex geometry can be learned from data.

3.2 *CoCoNet*

Figure 1 provides an overview of *CoCoNet* for learning convex conformity scores that induce efficient prediction regions. Given a pretrained predictor F, an encoder $\mathbf{h}_{\boldsymbol{\theta}}(.)$ maps the input $\mathbf{x}$ and its point prediction $\hat{\mathbf{y}} = F(\mathbf{x})$ to a contextual representation, which is then combined with the residual $\mathbf{r} = \mathbf{y} - F(\mathbf{x})$ by a structured scoring network to produce $S_{\boldsymbol{\theta}}(\mathbf{x}, \mathbf{r})$. The scoring network is designed to satisfy the *admissibility* conditions (6) required for convex prediction regions, while allowing their geometry to be learned from data. This architecture supports different convex score families, including the piecewise-linear Spline and the more flexible ICNN parameterizations introduced below. The parameters are learned on a disjoint score-learning set $\mathcal{D}_{\text{score}}$ using an efficiency-oriented surrogate objective. Once learned, the network is frozen, and split CP is applied on $\mathcal{D}_{\text{cal}}$ to compute the threshold $\widehat{\tau}_\alpha$, which is then used to construct valid prediction regions for new inputs. The following details the loss function for training and the architecture of *CoCoNet*.

Score Learning. To optimize prediction-region efficiency, *CoCoNet* learns the score parameters on the held-out dataset $\mathcal{D}_{\text{score}}$ using an empirical surrogate of the optimization in (5). The objective balances three goals: reducing prediction-region size, encouraging the desired coverage rate, and penalizing observations that fall outside the training sublevel region:

***CoCoNet* Score-Learning Objective**

$$\mathcal{L}(\boldsymbol{\theta}; \tau_{\text{tr}}) = \mathcal{L}_{\text{size}}(\boldsymbol{\theta}; \tau_{\text{tr}}) + \lambda_{\text{cov}} \mathcal{L}_{\text{cov}}(\boldsymbol{\theta}; \tau_{\text{tr}}) + \lambda_{\text{miss}} \mathcal{L}_{\text{miss}}(\boldsymbol{\theta}; \tau_{\text{tr}}). \tag{9}$$

The parameters of *CoCoNet* are learned using mini-batch optimization, where $\mathcal{B} = \{(\mathbf{x}_i, \mathbf{r}_i)\}_{i=1}^{|\mathcal{B}|}, \mathcal{B} \subseteq \mathcal{D}_{\text{score}}$.

The primary term directly promotes efficient prediction regions by minimizing their dimension-normalized average volume:

$$\mathcal{L}_{\text{size}}(\theta; \tau_{\text{tr}}) = \left[\frac{1}{|\mathcal{B}|} \sum_{i=1}^{|\mathcal{B}|} |\mathcal{C}_\theta(\mathbf{x}_i; \tau_{\text{tr}})| \right]^{1/p}. \tag{10}$$

The p-th root places the p-dimensional volume on a linear scale. Importantly, $\mathcal{L}_{\text{size}}$ acts directly on the regions induced by the learned score rather than fitting a predefined geometric template.

The remaining terms serve as soft constraints to maintain target coverage and prevent region collapse during training. $\mathcal{L}_{\text{cov}}$ penalizes deviations from the target inclusion rate $1 - \alpha$:

$$\mathcal{L}_{\text{cov}}(\boldsymbol{\theta}; \tau_{\text{tr}}) = \left[\widehat{\text{Cov}}_{\boldsymbol{\theta}} - (1 - \alpha) \right]^2, \quad \text{where} \quad \widehat{\text{Cov}}_{\boldsymbol{\theta}} = \frac{1}{|\mathcal{B}|} \sum_{i=1}^{|\mathcal{B}|} \sigma \left(\frac{\tau_{\text{tr}} - S_{\boldsymbol{\theta}}(\mathbf{x}_i)}{T} \right), \tag{11}$$

where $\sigma(\cdot)$ is the standard sigmoid function and $T > 0$ is a hyperparameter controlling smoothness.

Additionally, $\mathcal{L}_{\text{miss}}$ penalizes individual miscovered samples proportionally to their threshold violation magnitude:

$$\mathcal{L}_{\text{miss}}(\boldsymbol{\theta}; \tau_{\text{tr}}) = \frac{1}{|\mathcal{B}|} \sum_{i=1}^{|\mathcal{B}|} \text{ReLU} \left(\frac{S_{\boldsymbol{\theta}}(\mathbf{x}_i)}{\tau_{\text{tr}}} - 1 \right). \tag{12}$$

Thus, $\mathcal{L}_{\text{cov}}$ guides aggregate training coverage, while $\mathcal{L}_{\text{miss}}$ provides direct instance-level gradient signals. For simplicity of the training and optimization stability, the reference threshold is fixed to $\tau_{\text{tr}} = 1$. Once training completes, the scoring network parameters are frozen, and split CP is applied on an independent calibration set $\mathcal{D}_{\text{cal}}$ to derive the calibration threshold $\widehat{\tau}_\alpha$. This separation of score learning and calibration provides the finite-sample coverage guarantee under exchangeability.

The *CoCoNet* framework is agnostic to the specific architectural implementation of the conformity score, requiring only that $S_{\boldsymbol{\theta}}(\mathbf{x}, .) \in \mathcal{S}_{\text{adm}}$. Any learnable score family satisfying these structural admissibility conditions can seamlessly serve as the scoring module. The following instantiate two complementary score architectures: 1) a structured Spline parameterization, which yields input-adaptive convex sets with piecewise-linear boundaries, and 2) an ICNN-based approach, which provides high-capacity geometric flexibility.

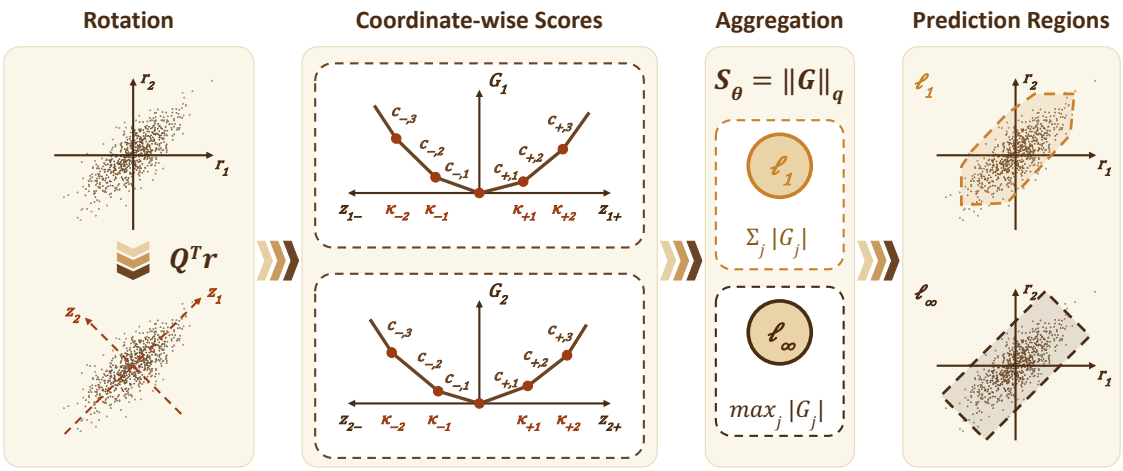

Figure 2: **Illustration of the learned rotation, coordinate-wise scores, and induced geometry.** The input-dependent transformation $\mathbf{z} = \mathbf{Q}^\top \mathbf{r}$ rotates the residual coordinates to align them with learned directions of variation (left). Each rotated coordinate is scored by convex piecewise-linear branches G_+ and G_- for positive and negative deviations, with knots $\kappa_{\pm,k}$ (center). The coordinate-wise scores are then aggregated with the ℓ_1 or ℓ_∞ norm: the ℓ_∞ aggregation induces a rotated hyperrectangle and the ℓ_1 aggregation an asymmetric polytope in the original residual space (right).

Feature Encoder. *CoCoNet* first encodes the input features and the predicted values using an encoder. This provides a contextual representation to the score, enabling *CoCoNet*'s score to adapt flexibly across the covariate space. In this work, the encoder is implemented as an MLP:

$$\mathbf{h}_{\boldsymbol{\theta}}(\mathbf{x}) = \mathrm{MLP}([\mathbf{x}, F(\mathbf{x})])\,. \tag{13}$$

Candidate residuals $\mathbf{r}$ bypass this encoder and enter the scoring head directly (Figure 1). This allows the score function to adapts its shape, orientation, and scale via the contextual vector $\mathbf{h}_{\boldsymbol{\theta}}(\mathbf{x})$ while maintaining strict convexity with respect to $\mathbf{r}$. Subscript $\boldsymbol{\theta}$ is omitted for simplicity in what follows.

CoCoNet$_{\text{Spline}}$**.** This instantiation parameterizes a piecewise-linear conformity score. Depending on the choice of a norm-based aggregation operator, *CoCoNet*$_{\text{Spline}}$ induces different classes of prediction regions including rotated hyperrectangulars or polytopes. Figure 2 shows an overview of *CoCoNet*$_{\text{Spline}}$ from rotation of the coordinates, to learning the splines across the coordinates, and aggregation via ℓ_1 and ℓ_∞.

Given context $\mathbf{h}(\mathbf{x})$ and $\mathbf{r} = \mathbf{y} - F(\mathbf{x})$, the residual vector is first mapped into a learned rotated coordinate space. The rotation is learned jointly with the conformity score to align the residual coordinates with input-dependent directions of variation, allowing the resulting prediction region to adapt its orientation to the local uncertainty structure:

$$\mathbf{z} = \mathbf{Q}^\top \mathbf{r} \in \mathbb{R}^p, \quad \text{where} \quad \mathbf{Q} = \exp(\mathbf{K}(\mathbf{h}(\mathbf{x})))\,. \tag{14}$$

Here, $\mathbf{K}(\mathbf{h}(\mathbf{x})) \in \mathbb{R}^{p\times p}$ is a skew-symmetric matrix ($\mathbf{K}^\top = -\mathbf{K}$) generated via a linear projection of $\mathbf{h}(\mathbf{x})$. Because the matrix exponential of a skew-symmetric matrix is strictly orthogonal, $\mathbf{Q}$ defines a smooth, input-dependent rotation of the residual space.

For each rotated coordinate $j \in \{1, \ldots, p\}$, independent piecewise-linear branches G_+ and G_- model positive and negative deviations. Denote K as the number of spline segments. Each branch anchors an initial knot at $\kappa_{\pm,0} = 0$, with subsequent knots $\kappa_{\pm,k}$ placed at empirical quantiles corresponding to levels $\{1/K, \ldots, 1\}$ of the coordinatewise residual magnitudes evaluated on $\mathcal{D}_{\text{score}}$. Let $a_{\pm,k}(\mathbf{h}(\mathbf{x}))$ denote the unconstrained neural network's output for the coordinate, $\pm$ branches, and segment $k \in \{1, \ldots, K\}$.

To guarantee convexity, each unconstrained network output is first transformed into a positive slope increment:

$$\tilde{c}_{\pm,k} = \mathrm{softplus}(a_{\pm,k}(\mathbf{h}(\mathbf{x}))) \tag{15}$$

The slope of segment k is then constructed by cumulatively summing these positive increments:

$$c_{\pm,k}(\mathbf{h}(\mathbf{x})) = \varepsilon_{\text{slope}} + \sum_{k=1}^{K} \tilde{c}_{\pm,k}, \qquad k = 1, \ldots, K, \tag{16}$$

Here, $\varepsilon_{\text{slope}} > 0$ ensures that all segment slopes remain strictly positive, preventing flat segments. Since each $\tilde{c}_{\pm,k} > 0$, the cumulative construction guarantees $c_{\pm,k+1} \geq c_{\pm,k}$, so the segment slopes are nondecreasing, which ensures convexity of the resulting piecewise-linear spline branch. The piecewise-constant function $c_{\pm}(t; \mathbf{h}(\mathbf{x})) = c_{\pm,k}(\mathbf{h}(\mathbf{x}))$ for $t \in [\kappa_{\pm,k-1}, \kappa_{\pm,k})$ therefore represents these learned segment slopes, the positive increments $\tilde{c}_{\pm,k}$ are first accumulated into nondecreasing slopes $c_{\pm,k}$, which are then integrated in (17) to construct the convex spline branch $G_{\pm}$.

$$G_{\pm}(s; \mathbf{h}(\mathbf{x})) = \int_0^s c_{\pm}(t; \mathbf{h}(\mathbf{x}))\, dt, \qquad s \geq 0. \tag{17}$$

By construction, $G_{\pm}(0) = 0$, and the strictly increasing sequence $0 < c_{\pm,1} \leq \cdots \leq c_{\pm,K}$ guarantees that each branch is convex and strictly increasing. Learning positive and negative branches independently allows *CoCoNet*$_{\text{Spline}}$ to capture asymmetric shapes. The same construction is applied independently to each rotated coordinate $j \in \mathcal{J}$. The conformity score is defined as a norm-based aggregation across the coordinates:

Spline Conformity Score

$$S_{\boldsymbol{\theta}}^{\text{Spline}}(\mathbf{x}, \mathbf{r}) = \left\| \left(G_{j,+}([z_j]_+; \mathbf{h}(\mathbf{x})) + G_{j,-}([-z_j]_+; \mathbf{h}(\mathbf{x})) \right)_{j=1}^{p} \right\|_q, \tag{18}$$

where $[a]_+ = \max(a, 0)$. This paper considers two aggregation norms, $q \in \{1, \infty\}$, which induce different convex prediction-region geometries. For ℓ_∞, the overall score is determined by the largest coordinatewise spline value, yielding asymmetric axis-aligned hyperrectangles in the rotated $\mathbf{z}$-space and input-adaptive rotated hyperrectangles in the original residual space. For ℓ_1, the coordinatewise spline scores are summed, inducing more flexible asymmetric convex polytopes with piecewise-linear boundaries. Since the coordinatewise spline functions are convex and nonnegative, both aggregation norms preserve convexity and satisfy $S_{\boldsymbol{\theta}}^{\text{Spline}} \in \mathcal{S}_{\text{adm}}$.

***CoCoNet*$_{\text{ICNN}}$.** To produce more complex and general convex prediction regions, *CoCoNet*$_{\text{ICNN}}$ parameterizes the score using an Input Convex Neural Network (Amos et al., 2017).

Let $\boldsymbol{\eta}_\ell \in \mathbb{R}^H$ denote the hidden state at layer ℓ. The network architecture is defined by:

$$\boldsymbol{\eta}_1 = \phi(\mathbf{A}_1(\mathbf{h}(\mathbf{x}))\mathbf{r} + \mathbf{V}_1\mathbf{h}(\mathbf{x}) + \mathbf{b}_1), \tag{19}$$

$$\boldsymbol{\eta}_\ell = \phi\big(\mathbf{W}_\ell^+ \boldsymbol{\eta}_{\ell-1} + \mathbf{A}_\ell(\mathbf{h}(\mathbf{x}))\mathbf{r} + \mathbf{V}_\ell\mathbf{h}(\mathbf{x}) + \mathbf{b}_\ell\big), \qquad \ell = 2, \ldots, L, \tag{20}$$

where $\mathbf{A}_\ell(\mathbf{h}(\mathbf{x}))\mathbf{r}$ term provides a learnable input-dependent linear transformation of the residual, $\mathbf{V}_\ell$ and $\mathbf{b}_\ell$ are learnable parameters, and $\phi(.)$ is a convex, non-decreasing activation function (softplus in this paper). Non-negativity constraints on the layer-to-layer weights ($\mathbf{W}_\ell^+ \geq \mathbf{0}$) ensure that the scalar output:

$$G_{\boldsymbol{\theta}}(\mathbf{x}, \mathbf{r}) = (\mathbf{w}^+)^\top \boldsymbol{\eta}_L \quad (\text{with } \mathbf{w}^+ \geq \mathbf{0}), \tag{21}$$

remains convex with respect to $\mathbf{r}$. To satisfy the admissibility conditions and non-negativity without violating convexity, the final score is constructed as:

ICNN Conformity Score

$$S_{\boldsymbol{\theta}}^{\text{ICNN}}(\mathbf{x}, \mathbf{r}) = G_{\boldsymbol{\theta}}(\mathbf{x}, \mathbf{r}) - G_{\boldsymbol{\theta}}(\mathbf{x}, \mathbf{0}) - \nabla_{\mathbf{r}} G_{\boldsymbol{\theta}}(\mathbf{x}, \mathbf{0})^\top \mathbf{r} + \varepsilon_{\text{ICNN}} \|\mathbf{r}\|_2^2. \tag{22}$$

The first three terms represent the Bregman divergence $D_G(\mathbf{r}, \mathbf{0})$ induced by $G_{\boldsymbol{\theta}}(\mathbf{x}, \cdot)$, guaranteeing non-negativity and $S_{\boldsymbol{\theta}}^{\text{ICNN}}(\mathbf{x}, \mathbf{0}) = 0$ by strict convexity. The small quadratic term ($\varepsilon_{\text{ICNN}} > 0$) enforces strong convexity, preventing unbounded or flat sublevel sets along unconstrained directions and ensuring $S_{\boldsymbol{\theta}}^{\text{ICNN}} \in \mathcal{S}_{\text{adm}}$. This formulation enables *CoCoNet*$_{\text{ICNN}}$ to learn highly flexible, arbitrary convex set geometries directly from multivariate uncertainty structures.

4 Experimental Setup

Datasets and Experimental Protocol. *CoCoNet* is evaluated on synthetic datasets with different residual geometries, and on 30 real-world multivariate regression tasks spanning diverse application domains, with response dimensions ranging from $p = 2$ to $p = 16$. All experiments are repeated across five random seeds, and results are reported as mean $\pm$ standard deviation. For each seed, the dataset is partitioned into four disjoint sets: $\mathcal{D}_{\text{pred}}$ (25%), $\mathcal{D}_{\text{score}}$ (30%), $\mathcal{D}_{\text{cal}}$ (25%), and $\mathcal{D}_{\text{test}}$ (20%). A common MLP base predictor is trained on $\mathcal{D}_{\text{pred}}$ then frozen, so all CP methods use the same predictions and residuals. $\mathcal{D}_{\text{score}}$ is used for score learning and hyperparameter tuning, $\mathcal{D}_{\text{cal}}$ for split conformal threshold calibration, and $\mathcal{D}_{\text{test}}$ for held-out evaluation. Dataset specifications and implementation details appear in Appendix D.

Baselines. *CoCoNet* is benchmarked against five representative multi-dimensional CP approaches: (1) **Absolute-Residual CP** (Lei et al., 2018), which uses coordinatewise max-residual quantiles to form axis-aligned hyperrectangles; (2) **Empirical Copula CP** (Messoudi et al., 2021), which models dependence among coordinatewise residuals through an empirical copula; (3) **Global Ellipsoid** (Johnstone and Cox, 2021), constructing static ellipsoidal regions based on a global covariance Mahalanobis score; (4) **Local Ellipsoid** (Messoudi et al., 2022), which uses input-dependent covariance estimates to produce localized ellipsoidal regions; and (5) **Minimum Volume Conformal Sets (MVCS)** (Braun et al., 2026), which learns input-adaptive norm-based regions. The single-norm variant centered at the base prediction with norm exponent $q \geq 1$ to guarantee convexity is used.

All methods target 90% nominal marginal coverage ($1 - \alpha = 0.90$) and are evaluated on empirical coverage rate and dimension-normalized volume $|\mathcal{C}(\mathbf{x})|^{1/p}$. Identical base predictors and data splits are used across all methods. For the synthetic experiments, the evaluation also includes Intersection over Union (IoU) and the volume ratio relative to the population-optimal region ($V/V^\star$). Formal definitions of all evaluation metrics are provided in Appendix D.

5 Results

Synthetic Geometry Recovery. To evaluate geometric adaptivity across diverse structures, four two-dimensional synthetic residual distributions with known population-optimal 90% minimum-area regions are considered: an ellipsoid, a rectangle, a triangle, and a trapezoid. These target geometries span standard quadric and polyhedral families alongside asymmetric, non-standard polygonal shapes demanding uncon-

Table 1: **Recovery of population-optimal** 90% **convex regions.** IoU and $V/V^\star$ compare the learned region with the population minimum-area region. Subscripts are standard deviations over five seeds; best in **blue**, second-best underlined. Gray rows report the empirical test coverage in percent.

		Baselines					*CoCoNet*		
Shape	Metric	Abs. Res.	Emp. Cop.	G-Ellip.	L-Ellip.	MVCS	Spline-ℓ_∞	Spline-ℓ_1	ICNN
Ellipsoid	IoU↑	$0.35_{\pm 0.00}$	$0.79_{\pm 0.01}$	$\mathbf{0.97_{\pm 0.01}}$	$0.88_{\pm 0.02}$	$\mathbf{0.97_{\pm 0.02}}$	$0.79_{\pm 0.01}$	$0.83_{\pm 0.03}$	$0.94_{\pm 0.01}$
	$V/V^\star$↓	$2.67_{\pm 0.03}$	$1.23_{\pm 0.01}$	$\mathbf{1.03_{\pm 0.01}}$	$1.12_{\pm 0.02}$	$\mathbf{1.03_{\pm 0.02}}$	$1.22_{\pm 0.01}$	$1.18_{\pm 0.04}$	$1.06_{\pm 0.02}$
	Cov. %	$89.5_{\pm 0.9}$	$90.6_{\pm 1.1}$	$90.7_{\pm 1.6}$	$90.6_{\pm 1.5}$	$90.6_{\pm 1.7}$	$90.6_{\pm 1.4}$	$90.7_{\pm 1.7}$	$90.5_{\pm 1.5}$
Rectangle	IoU↑	$0.53_{\pm 0.01}$	$0.57_{\pm 0.01}$	$0.75_{\pm 0.01}$	$0.73_{\pm 0.01}$	$0.76_{\pm 0.01}$	$\mathbf{0.90_{\pm 0.02}}$	$0.70_{\pm 0.01}$	$\underline{0.86_{\pm 0.01}}$
	$V/V^\star$↓	$1.80_{\pm 0.02}$	$1.68_{\pm 0.03}$	$1.29_{\pm 0.02}$	$1.33_{\pm 0.02}$	$1.29_{\pm 0.02}$	$\mathbf{1.10_{\pm 0.02}}$	$1.37_{\pm 0.02}$	$\underline{1.15_{\pm 0.01}}$
	Cov. %	$90.7_{\pm 1.3}$	$90.7_{\pm 1.4}$	$90.5_{\pm 1.5}$	$90.5_{\pm 1.0}$	$90.6_{\pm 1.4}$	$90.6_{\pm 1.2}$	$90.5_{\pm 1.5}$	$90.7_{\pm 1.0}$
Triangle	IoU↑	$0.45_{\pm 0.01}$	$0.51_{\pm 0.01}$	$0.57_{\pm 0.01}$	$0.56_{\pm 0.01}$	$0.57_{\pm 0.00}$	$0.58_{\pm 0.01}$	$\underline{0.64_{\pm 0.03}}$	$\mathbf{0.85_{\pm 0.02}}$
	$V/V^\star$↓	$2.05_{\pm 0.05}$	$1.82_{\pm 0.03}$	$1.62_{\pm 0.02}$	$1.65_{\pm 0.03}$	$1.64_{\pm 0.01}$	$1.61_{\pm 0.03}$	$\underline{1.47_{\pm 0.06}}$	$\mathbf{1.15_{\pm 0.02}}$
	Cov. %	$90.5_{\pm 1.1}$	$90.1_{\pm 1.3}$	$90.4_{\pm 1.7}$	$90.1_{\pm 1.8}$	$90.3_{\pm 1.7}$	$90.3_{\pm 1.6}$	$90.5_{\pm 1.8}$	$90.7_{\pm 1.7}$
Trapezoid	IoU↑	$0.56_{\pm 0.01}$	$0.56_{\pm 0.01}$	$0.67_{\pm 0.01}$	$0.65_{\pm 0.01}$	$0.68_{\pm 0.01}$	$0.71_{\pm 0.01}$	$\underline{0.73_{\pm 0.03}}$	$\mathbf{0.83_{\pm 0.02}}$
	$V/V^\star$↓	$1.66_{\pm 0.02}$	$1.66_{\pm 0.02}$	$1.42_{\pm 0.02}$	$1.45_{\pm 0.01}$	$1.40_{\pm 0.01}$	$1.35_{\pm 0.02}$	$\underline{1.32_{\pm 0.06}}$	$\mathbf{1.19_{\pm 0.02}}$
	Cov. %	$90.6_{\pm 1.6}$	$90.3_{\pm 1.6}$	$90.5_{\pm 1.7}$	$90.5_{\pm 1.7}$	$90.3_{\pm 1.9}$	$90.1_{\pm 1.6}$	$90.6_{\pm 1.6}$	$89.9_{\pm 1.5}$

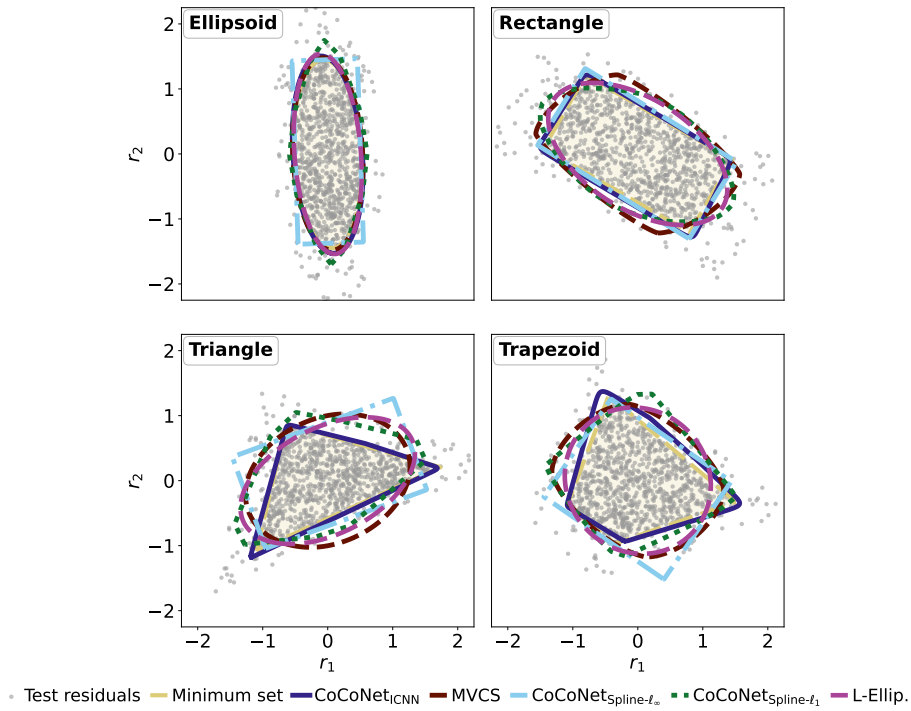

Figure 3: **Synthetic uncertainty geometries.** Calibrated 90% prediction regions are shown for four two-dimensional residual distributions with known population-optimal regions: an ellipsoid, rectangle, triangle, and trapezoid. The comparison illustrates the progression from geometries well matched by structured families to more challenging asymmetric polygonal shapes, where the greater geometric flexibility of *CoCoNet*$_{\text{ICNN}}$ becomes increasingly important.

strained boundary flexibility. Table 1 reports empirical coverage, IoU, and relative volume, while Figure 3 visualizes the recovered prediction regions. Synthetic data construction and implementation details are provided in Appendix D.

The empirical findings demonstrate that baseline approaches perform effectively when the ground-truth uncertainty matches their predefined geometric templates but their efficiency degrades sharply when target geometries depart from those assumptions. On the ellipsoidal target, MVCS and Ellipsoidal CP achieve high IoUs of 0.97 and 0.96, respectively; however, their IoU performance drops by 0.3–0.4 on the triangle and trapezoid targets. On the rectangular target,*CoCoNet*$_{\text{Spline-}\ell_\infty}$ achieves the highest IoU of 0.90, matching its rotated hyperrectangular parameterization, but loses efficiency on non-rectangular geometries. Replacing its ℓ_∞ aggregation with the ℓ_1 norm yields asymmetric polygonal regions: *CoCoNet*$_{\text{Spline-}\ell_1}$ ranks second on the triangle and trapezoid targets, with IoUs of 0.64 and 0.73, but it falls behind MVCS and Ellipsoidal CP on the ellipsoidal and rectangular targets. In contrast, *CoCoNet*$_{\text{ICNN}}$ dynamically adjusts to different underlying geometries, maintaining consistently high IoU scores across all evaluated scenarios.

Overall, the three *CoCoNet* variants achieve first- or second-best performance across three of the four target geometries while remaining highly competitive on the ellipsoidal setting. This highlights *CoCoNet*'s adaptivity in learning flexible conformity scores that consistently induce efficient prediction regions across diverse geometries.

Real-World Benchmarks. Table 2 reports the dimension-normalized prediction-region size across 30 real-world multivariate regression tasks with response dimensions ranging from $p = 2$ to $p = 16$. To facilitate aggregate comparison across all methods and datasets, mean efficiency ranks are also reported, where lower values indicate superior prediction-region efficiency. Additional analysis and qualitative examples are provided in Appendix E. The results show a broad improvement as the score families gain flexibility to

Table 2: **Real-world multivariate regression results at 90% target coverage.** Each method reports, in gray, the empirical test coverage in percent, followed by the dimension-normalized prediction-region size (lower is better); subscripts are standard deviations over five seeds. Where a factor ($\times 10^k$) follows the dataset name, the sizes in that row are in units of 10^k. Best and second-best sizes are shown in **blue** and underline

Dataset	p	Baselines										*CoCoNet*					
		Abs. Res.		Emp. Cop.		G-Ellip.		L-Ellip.		MVCS		Spline-ℓ_∞		Spline-ℓ_1		ICNN	
		Cov. %	Size	Cov. %	Size	Cov. %	Size	Cov. %	Size	Cov. %	Size	Cov. %	Size	Cov. %	Size	Cov. %	Size
Argoverse2-2D ($\times 10^1$)	2	$89.9_{\pm 0.8}$	$3.98_{\pm 0.05}$	$89.8_{\pm 0.6}$	$3.73_{\pm 0.05}$	$89.9_{\pm 0.5}$	$3.86_{\pm 0.03}$	$90.1_{\pm 0.5}$	$3.67_{\pm 0.05}$	$90.2_{\pm 0.7}$	$3.68_{\pm 0.08}$	$90.1_{\pm 0.6}$	$3.58_{\pm 0.03}$	$90.2_{\pm 0.7}$	$\mathbf{3.50_{\pm 0.10}}$	$89.8_{\pm 0.6}$	$3.51_{\pm 0.06}$
Argoverse2 Motion ($\times 10^1$)	2	$90.1_{\pm 0.8}$	$1.23_{\pm 0.03}$	$90.0_{\pm 0.8}$	$1.16_{\pm 0.03}$	$90.3_{\pm 0.6}$	$1.20_{\pm 0.02}$	$89.9_{\pm 0.8}$	$1.14_{\pm 0.02}$	$90.0_{\pm 0.7}$	$1.14_{\pm 0.03}$	$90.1_{\pm 0.6}$	$1.11_{\pm 0.02}$	$90.3_{\pm 0.9}$	$1.11_{\pm 0.03}$	$89.9_{\pm 0.6}$	$\mathbf{1.10_{\pm 0.01}}$
Bias-Corr. Temp.	2	$90.0_{\pm 0.9}$	$4.25_{\pm 0.08}$	$89.8_{\pm 0.9}$	$4.16_{\pm 0.05}$	$89.8_{\pm 0.9}$	$4.08_{\pm 0.06}$	$90.1_{\pm 0.7}$	$3.98_{\pm 0.06}$	$89.8_{\pm 0.8}$	$4.08_{\pm 0.08}$	$89.7_{\pm 0.8}$	$3.86_{\pm 0.10}$	$89.9_{\pm 0.9}$	$\mathbf{3.82_{\pm 0.13}}$	$90.4_{\pm 0.5}$	$3.88_{\pm 0.13}$
ENB	2	$88.8_{\pm 2.8}$	$12.76_{\pm 1.32}$	$88.7_{\pm 4.4}$	$10.24_{\pm 0.95}$	$89.7_{\pm 4.9}$	$10.94_{\pm 1.35}$	$90.9_{\pm 2.7}$	$9.19_{\pm 0.58}$	$90.0_{\pm 2.5}$	$9.56_{\pm 2.32}$	$90.3_{\pm 3.0}$	$8.14_{\pm 1.24}$	$90.5_{\pm 1.8}$	$8.19_{\pm 0.39}$	$89.4_{\pm 3.5}$	$\mathbf{7.55_{\pm 0.13}}$
Geographical Music ($\times 10^2$)	2	$90.6_{\pm 1.8}$	$1.14_{\pm 0.03}$	$89.2_{\pm 0.8}$	$1.31_{\pm 0.20}$	$90.1_{\pm 1.9}$	$1.11_{\pm 0.03}$	$89.7_{\pm 1.3}$	$1.16_{\pm 0.05}$	$89.1_{\pm 2.6}$	$1.26_{\pm 0.15}$	$90.7_{\pm 1.6}$	$\mathbf{1.06_{\pm 0.02}}$	$91.5_{\pm 2.1}$	$1.08_{\pm 0.04}$	$91.2_{\pm 2.0}$	$1.14_{\pm 0.13}$
CASP ($\times 10^1$)	2	$90.5_{\pm 0.3}$	$3.87_{\pm 0.08}$	$90.2_{\pm 0.5}$	$2.80_{\pm 0.15}$	$90.3_{\pm 0.5}$	$3.01_{\pm 0.02}$	$90.3_{\pm 0.6}$	$2.87_{\pm 0.05}$	$89.8_{\pm 0.6}$	$2.48_{\pm 0.08}$	$89.7_{\pm 0.9}$	$2.42_{\pm 0.06}$	$89.7_{\pm 0.4}$	$2.39_{\pm 0.08}$	$89.8_{\pm 0.8}$	$\mathbf{2.34_{\pm 0.09}}$
Online Retail II ($\times 10^1$)	2	$90.1_{\pm 0.3}$	$3.26_{\pm 0.46}$	$90.0_{\pm 0.6}$	$2.60_{\pm 0.13}$	$89.8_{\pm 0.6}$	$4.62_{\pm 2.17}$	$89.8_{\pm 0.3}$	$5.94_{\pm 2.10}$	$89.9_{\pm 0.4}$	$2.25_{\pm 0.12}$	$89.7_{\pm 0.4}$	$2.05_{\pm 0.06}$	$89.8_{\pm 0.6}$	$1.92_{\pm 0.11}$	$90.0_{\pm 0.5}$	$\mathbf{1.85_{\pm 0.12}}$
QM9-2D	2	$89.9_{\pm 0.7}$	$8.79_{\pm 0.13}$	$89.2_{\pm 0.8}$	$5.40_{\pm 0.10}$	$89.5_{\pm 0.5}$	$5.73_{\pm 0.06}$	$89.6_{\pm 0.5}$	$5.36_{\pm 0.11}$	$89.8_{\pm 0.5}$	$5.30_{\pm 0.06}$	$89.6_{\pm 0.4}$	$\mathbf{5.06_{\pm 0.07}}$	$89.4_{\pm 0.4}$	$5.10_{\pm 0.10}$	$89.5_{\pm 0.8}$	$5.08_{\pm 0.11}$
Household Power	3	$90.3_{\pm 0.3}$	$22.44_{\pm 0.62}$	$90.3_{\pm 0.4}$	$16.09_{\pm 1.42}$	$90.2_{\pm 0.3}$	$23.31_{\pm 1.09}$	$90.1_{\pm 0.3}$	$15.73_{\pm 0.40}$	$89.9_{\pm 0.5}$	$5.46_{\pm 0.37}$	$90.1_{\pm 0.4}$	$7.15_{\pm 0.71}$	$90.2_{\pm 0.4}$	$6.96_{\pm 0.86}$	$89.9_{\pm 0.4}$	$\mathbf{5.10_{\pm 0.66}}$
SCPF	3	$89.9_{\pm 1.4}$	$7.32_{\pm 0.32}$	$89.6_{\pm 1.6}$	$6.29_{\pm 0.73}$	$89.4_{\pm 2.6}$	$7.63_{\pm 1.16}$	$89.9_{\pm 2.2}$	$7.27_{\pm 0.92}$	$90.9_{\pm 1.1}$	$5.82_{\pm 0.32}$	$89.6_{\pm 1.5}$	$6.07_{\pm 0.99}$	$88.2_{\pm 3.1}$	$5.19_{\pm 0.80}$	$89.0_{\pm 3.2}$	$\mathbf{5.16_{\pm 0.68}}$
Argoverse2-4D ($\times 10^1$)	4	$90.1_{\pm 0.8}$	$2.86_{\pm 0.06}$	$90.0_{\pm 0.6}$	$2.52_{\pm 0.04}$	$90.1_{\pm 0.4}$	$1.64_{\pm 0.02}$	$90.0_{\pm 0.4}$	$1.59_{\pm 0.02}$	$90.1_{\pm 0.5}$	$1.61_{\pm 0.02}$	$89.9_{\pm 0.3}$	$1.64_{\pm 0.02}$	$90.2_{\pm 0.4}$	$1.60_{\pm 0.04}$	$89.9_{\pm 0.6}$	$\mathbf{1.57_{\pm 0.02}}$
QM8 ($\times 10^{-2}$)	4	$89.8_{\pm 0.8}$	$19.13_{\pm 0.53}$	$89.9_{\pm 0.8}$	$16.91_{\pm 0.48}$	$89.7_{\pm 0.8}$	$15.32_{\pm 0.27}$	$90.0_{\pm 1.0}$	$13.76_{\pm 0.17}$	$89.4_{\pm 0.6}$	$11.89_{\pm 0.09}$	$89.7_{\pm 0.8}$	$11.37_{\pm 0.23}$	$89.6_{\pm 0.3}$	$11.07_{\pm 0.09}$	$89.3_{\pm 0.8}$	$\mathbf{9.66_{\pm 0.21}}$
QM9-4D	4	$89.5_{\pm 0.6}$	$5.06_{\pm 0.07}$	$89.3_{\pm 0.7}$	$3.94_{\pm 0.03}$	$89.5_{\pm 0.8}$	$3.71_{\pm 0.03}$	$89.5_{\pm 0.8}$	$3.55_{\pm 0.06}$	$89.6_{\pm 0.5}$	$3.48_{\pm 0.04}$	$89.8_{\pm 0.2}$	$3.55_{\pm 0.04}$	$89.6_{\pm 0.8}$	$3.47_{\pm 0.03}$	$89.9_{\pm 0.6}$	$\mathbf{3.34_{\pm 0.02}}$
SGEMM ($\times 10^1$)	4	$90.0_{\pm 0.9}$	$12.50_{\pm 0.43}$	$89.9_{\pm 1.0}$	$13.05_{\pm 0.75}$	$90.1_{\pm 0.4}$	$3.58_{\pm 0.12}$	$90.1_{\pm 0.4}$	$3.51_{\pm 0.06}$	$89.9_{\pm 0.5}$	$3.03_{\pm 0.17}$	$90.1_{\pm 0.4}$	$3.22_{\pm 0.20}$	$90.1_{\pm 0.6}$	$3.08_{\pm 0.16}$	$90.1_{\pm 0.4}$	$\mathbf{3.02_{\pm 0.18}}$
Beijing AQ-6D ($\times 10^2$)	6	$89.7_{\pm 0.6}$	$3.04_{\pm 0.03}$	$89.8_{\pm 0.4}$	$3.08_{\pm 0.07}$	$89.7_{\pm 0.5}$	$2.14_{\pm 0.02}$	$89.7_{\pm 0.7}$	$2.03_{\pm 0.02}$	$89.7_{\pm 0.5}$	$1.90_{\pm 0.03}$	$89.9_{\pm 0.5}$	$1.97_{\pm 0.02}$	$89.9_{\pm 0.3}$	$1.82_{\pm 0.01}$	$89.9_{\pm 0.6}$	$\mathbf{1.68_{\pm 0.02}}$
Communities Crime ($\times 10^2$)	6	$90.6_{\pm 3.6}$	$7.41_{\pm 0.42}$	$89.7_{\pm 1.7}$	$8.43_{\pm 0.57}$	$91.3_{\pm 1.5}$	$6.23_{\pm 0.20}$	$90.8_{\pm 2.6}$	$6.81_{\pm 0.29}$	$91.7_{\pm 1.1}$	$6.30_{\pm 0.35}$	$91.6_{\pm 1.7}$	$6.94_{\pm 0.54}$	$90.6_{\pm 2.2}$	$6.07_{\pm 0.10}$	$90.6_{\pm 1.7}$	$\mathbf{5.94_{\pm 0.17}}$
ATP1D ($\times 10^2$)	6	$90.0_{\pm 3.9}$	$5.19_{\pm 1.08}$	$82.6_{\pm 5.2}$	$6.12_{\pm 1.05}$	$91.5_{\pm 3.0}$	$3.73_{\pm 0.51}$	$89.7_{\pm 3.8}$	$4.44_{\pm 0.51}$	$89.4_{\pm 3.1}$	$4.61_{\pm 1.07}$	$89.4_{\pm 5.5}$	$4.20_{\pm 0.47}$	$88.5_{\pm 4.9}$	$4.50_{\pm 0.91}$	$88.8_{\pm 3.8}$	$\mathbf{3.60_{\pm 0.43}}$
QM9-6D	6	$89.7_{\pm 0.8}$	$10.57_{\pm 0.08}$	$89.7_{\pm 0.6}$	$6.17_{\pm 0.10}$	$89.5_{\pm 0.8}$	$5.46_{\pm 0.06}$	$89.7_{\pm 1.0}$	$5.21_{\pm 0.08}$	$90.0_{\pm 0.7}$	$5.07_{\pm 0.06}$	$90.0_{\pm 0.5}$	$5.35_{\pm 0.07}$	$89.8_{\pm 1.0}$	$4.90_{\pm 0.08}$	$90.1_{\pm 0.6}$	$\mathbf{4.85_{\pm 0.11}}$
SARCOS	7	$90.2_{\pm 0.7}$	$5.56_{\pm 0.09}$	$89.8_{\pm 0.6}$	$5.19_{\pm 0.21}$	$89.9_{\pm 0.8}$	$4.16_{\pm 0.07}$	$90.5_{\pm 0.8}$	$3.90_{\pm 0.03}$	$89.7_{\pm 0.5}$	$3.61_{\pm 0.05}$	$89.9_{\pm 0.8}$	$3.87_{\pm 0.07}$	$89.9_{\pm 0.4}$	$3.48_{\pm 0.09}$	$89.8_{\pm 0.4}$	$\mathbf{3.34_{\pm 0.01}}$
Argoverse2-8D	8	$90.1_{\pm 0.6}$	$22.45_{\pm 0.22}$	$89.7_{\pm 0.6}$	$26.27_{\pm 4.20}$	$89.9_{\pm 0.4}$	$6.90_{\pm 0.15}$	$89.8_{\pm 0.4}$	$6.81_{\pm 0.09}$	$90.1_{\pm 0.4}$	$6.84_{\pm 0.11}$	$90.0_{\pm 0.8}$	$7.83_{\pm 0.24}$	$90.0_{\pm 0.7}$	$6.96_{\pm 0.12}$	$90.1_{\pm 0.6}$	$\mathbf{6.76_{\pm 0.10}}$
QM9-8D	8	$89.7_{\pm 0.7}$	$12.17_{\pm 0.14}$	$89.7_{\pm 0.5}$	$6.03_{\pm 0.10}$	$89.8_{\pm 1.2}$	$4.98_{\pm 0.11}$	$89.7_{\pm 0.8}$	$4.77_{\pm 0.08}$	$90.2_{\pm 0.8}$	$4.62_{\pm 0.09}$	$89.9_{\pm 0.5}$	$5.06_{\pm 0.09}$	$89.9_{\pm 0.7}$	$\mathbf{4.38_{\pm 0.09}}$	$90.1_{\pm 0.9}$	$4.41_{\pm 0.09}$
Beijing AQ-12D ($\times 10^2$)	12	$89.7_{\pm 0.7}$	$2.87_{\pm 0.04}$	$89.6_{\pm 0.3}$	$4.32_{\pm 0.21}$	$89.8_{\pm 0.4}$	$1.34_{\pm 0.03}$	$89.4_{\pm 0.5}$	$1.52_{\pm 0.03}$	$89.7_{\pm 0.3}$	$1.34_{\pm 0.03}$	$89.5_{\pm 0.2}$	$1.55_{\pm 0.02}$	$89.7_{\pm 0.6}$	$\mathbf{1.31_{\pm 0.01}}$	$89.4_{\pm 0.3}$	$1.33_{\pm 0.05}$
Knee Contact Force ($\times 10^1$)	12	$89.8_{\pm 2.3}$	$19.13_{\pm 0.93}$	$90.1_{\pm 1.4}$	$26.66_{\pm 3.47}$	$89.6_{\pm 1.9}$	$7.68_{\pm 0.26}$	$89.2_{\pm 0.9}$	$8.43_{\pm 0.20}$	$90.1_{\pm 1.8}$	$6.81_{\pm 0.06}$	$90.4_{\pm 1.2}$	$8.27_{\pm 0.22}$	$89.8_{\pm 1.6}$	$7.16_{\pm 0.11}$	$90.3_{\pm 2.3}$	$\mathbf{6.65_{\pm 0.33}}$
WQ	14	$89.6_{\pm 2.9}$	$8.60_{\pm 0.45}$	$85.3_{\pm 2.5}$	$10.32_{\pm 0.73}$	$90.0_{\pm 1.7}$	$5.99_{\pm 0.14}$	$88.8_{\pm 3.5}$	$6.76_{\pm 0.42}$	$89.4_{\pm 2.5}$	$\mathbf{5.95_{\pm 0.27}}$	$89.0_{\pm 1.8}$	$6.93_{\pm 0.19}$	$89.3_{\pm 1.4}$	$6.28_{\pm 0.53}$	$87.9_{\pm 2.1}$	$6.11_{\pm 0.26}$
QM7b	14	$90.0_{\pm 1.3}$	$5.30_{\pm 0.13}$	$90.0_{\pm 1.2}$	$5.98_{\pm 0.62}$	$90.4_{\pm 0.8}$	$2.14_{\pm 0.08}$	$90.4_{\pm 0.3}$	$2.18_{\pm 0.09}$	$90.4_{\pm 0.7}$	$1.88_{\pm 0.03}$	$90.5_{\pm 0.7}$	$2.32_{\pm 0.02}$	$89.9_{\pm 1.2}$	$1.97_{\pm 0.03}$	$90.1_{\pm 0.6}$	$\mathbf{1.78_{\pm 0.03}}$
OES10 ($\times 10^3$)	16	$91.2_{\pm 4.7}$	$3.09_{\pm 0.69}$	$85.5_{\pm 5.8}$	$6.17_{\pm 1.17}$	$89.8_{\pm 5.1}$	$2.09_{\pm 0.31}$	$90.0_{\pm 2.9}$	$\mathbf{2.01_{\pm 0.15}}$	$90.2_{\pm 5.4}$	$2.77_{\pm 0.68}$	$90.8_{\pm 4.7}$	$3.86_{\pm 1.13}$	$90.5_{\pm 4.4}$	$2.95_{\pm 0.64}$	$91.0_{\pm 4.8}$	$2.39_{\pm 0.51}$
OES97 ($\times 10^3$)	16	$90.0_{\pm 3.9}$	$4.24_{\pm 0.62}$	$83.9_{\pm 3.4}$	$15.32_{\pm 4.57}$	$90.6_{\pm 6.4}$	$2.97_{\pm 0.27}$	$90.6_{\pm 1.8}$	$3.60_{\pm 0.44}$	$92.4_{\pm 4.7}$	$4.85_{\pm 1.16}$	$90.0_{\pm 3.9}$	$4.63_{\pm 0.93}$	$91.5_{\pm 3.9}$	$3.16_{\pm 0.42}$	$92.7_{\pm 2.2}$	$\mathbf{2.87_{\pm 0.29}}$
BDG2-Fox ($\times 10^1$)	16	$89.7_{\pm 0.6}$	$14.13_{\pm 0.35}$	$89.6_{\pm 0.4}$	$18.73_{\pm 2.86}$	$89.7_{\pm 0.4}$	$9.09_{\pm 0.10}$	$89.7_{\pm 0.8}$	$8.33_{\pm 0.09}$	$89.5_{\pm 0.9}$	$7.72_{\pm 0.11}$	$89.9_{\pm 0.7}$	$9.73_{\pm 0.31}$	$89.7_{\pm 0.8}$	$10.97_{\pm 1.65}$	$89.7_{\pm 1.2}$	$\mathbf{7.54_{\pm 0.16}}$
BDG2-Eagle ($\times 10^1$)	16	$90.1_{\pm 0.9}$	$11.46_{\pm 0.17}$	$89.8_{\pm 0.6}$	$12.29_{\pm 0.56}$	$90.3_{\pm 0.6}$	$7.77_{\pm 0.07}$	$89.6_{\pm 0.7}$	$7.29_{\pm 0.09}$	$89.8_{\pm 0.5}$	$7.15_{\pm 0.06}$	$89.4_{\pm 0.8}$	$8.99_{\pm 0.16}$	$89.7_{\pm 0.4}$	$7.35_{\pm 0.08}$	$89.8_{\pm 1.1}$	$\mathbf{7.13_{\pm 0.12}}$
BDG2-Lamb ($\times 10^1$)	16	$89.6_{\pm 0.6}$	$17.64_{\pm 0.16}$	$89.5_{\pm 0.6}$	$23.77_{\pm 2.51}$	$89.4_{\pm 0.2}$	$10.15_{\pm 0.06}$	$89.4_{\pm 0.7}$	$8.80_{\pm 0.10}$	$89.6_{\pm 0.7}$	$7.31_{\pm 0.12}$	$90.2_{\pm 0.6}$	$8.95_{\pm 0.20}$	$89.6_{\pm 0.4}$	$7.51_{\pm 0.15}$	$89.6_{\pm 0.7}$	$\mathbf{7.19_{\pm 0.25}}$
Mean rank ↓		7.18 ± 0.88		7.13 ± 1.02		5.15 ± 1.82		4.47 ± 1.35		3.58 ± 1.48		4.28 ± 1.62		2.75 ± 1.26		$\mathbf{1.45 \pm 0.86}$	

represent joint uncertainty. Absolute-Residual CP and Empirical Copula have highest mean ranks of 7.18 and 7.13, respectively. Global and Local Ellipsoid improve to 5.15 and 4.47 by accounting for cross-output covariance and orientation; the local variant also adapts this structure to the input which results in better overall performance. MVCS is the strongest baseline, with a mean rank of 3.58, highlighting its flexibility through norm-based and input adaptive conformity score modeling. *CoCoNet*$_{\text{Spline-}\ell_\infty}$ achieves a mean rank of 4.28: its input-dependent rotation and asymmetric widths are effective for box-like uncertainty, although its regions remain rotated asymmetric hyperrectangles. While *CoCoNet*$_{\text{Spline-}\ell_\infty}$ falls behind MVCS, it outperform the other baselines and produce efficient linear prediction regions. Replacing its ℓ_∞ aggregation with the ℓ_1 norm, *CoCoNet*$_{\text{Spline-}\ell_1}$ improves the mean rank to 2.75, ahead of MVCS, and is the most efficient method on 4 tasks. Finally, *CoCoNet*$_{\text{ICNN}}$ achieves the best mean rank of 1.45. It is the most efficient method on 22 of 30 tasks, ranks among the top two on 26.

The gains of *CoCoNet*$_{\text{ICNN}}$ span response dimensions and application domains. Relative to the next-most efficient method on each task, its dimension-normalized size is approximately 3.6% smaller on Online Retail II, 7.7% smaller on Beijing AQ-6D, 4.0% smaller on SARCOS, and 5.3% smaller on QM7b. Note that this metric is the p-th root of the prediction-region volume and these differences correspond to substantially larger reductions in the original p-dimensional volume. It translates to approximately 7%, 38%, 25%, and 53% for Online Retail II, Beijing AQ-6D, SARCOS, and QM7b, respectively. For $p \geq 14$, *CoCoNet*$_{\text{ICNN}}$ is the most efficient method on QM7b and all three 16-dimensional Building Data Genome tasks. The more structured MVCS and Local Ellipsoid perform better on the smaller WQ and OES10 datasets, although *CoCoNet*$_{\text{ICNN}}$ also leads on the small OES97 dataset. Figure 9 summarizes rank distributions across the 30 multivariate tasks.

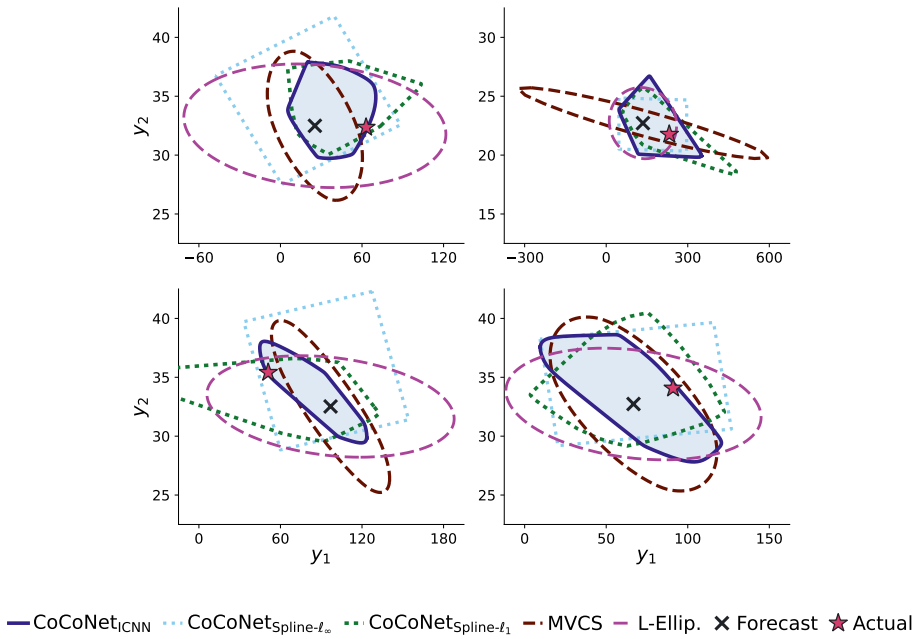

Figure 4: **Input-adaptive prediction-region geometry on CASP.** Four test instances compare *CoCoNet*$_{\text{ICNN}}$ with Local Ellipsoid, MVCS, and *CoCoNet*$_{\text{Spline-}\ell_\infty}$ and *CoCoNet*$_{\text{Spline-}\ell_1}$, using the same fixed point predictor. The cross denotes the point prediction and the star denotes the observed response. Local Ellipsoid adapts orientation and scale while retaining an elliptical boundary; MVCS, *CoCoNet*$_{\text{Spline-}\ell_\infty}$, and *CoCoNet*$_{\text{Spline-}\ell_1}$ adapt within their respective norm-based, rotated-hyperrectangular, and polytope families. In contrast, *CoCoNet*$_{\text{ICNN}}$ can also adapt the boundary geometry itself, producing qualitatively different convex regions across inputs.

Further analysis in Appendix E reports univariate performance ($p = 1$; Table 7), and extends the comparison to potentially non-convex MVCS variants and Pullback Conformal Sets (PBS) Kondratyev et al. (2025).

Figure 4 shows examples of prediction regions across four representative instances of the CASP dataset, presenting the results of the five top-performing CP approaches for qualitative comparison. While Local Ellipsoid, MVCS, and the two spline variants adjust their prediction regions strictly within predefined geometric families, *CoCoNet*$_{\text{ICNN}}$ dynamically adapts the boundary geometry itself across different inputs, producing regions with distinct orientations, asymmetries, and shapes tailored to local residual structures.

6 Conclusion

This paper introduced *CoCoNet*, a framework for learning convex conformity scores that induce flexible, data-adaptive prediction regions for multivariate regression settings. By defining an admissible class of conformity scores $\mathcal{S}_{\text{adm}}$ that strictly guarantees regional convexity, *CoCoNet* parameterizes and optimizes score functions for maximal region efficiency using held-out data. Two distinct architectures were developed: a structured Spline variant combining input-dependent space rotations with asymmetric coordinatewise functions, and an ICNN-based variant capable of parameterizing general convex geometries. Experiments across synthetic tasks and 30 real-world multivariate benchmarks (with output dimension $p \in [2, 16]$) demonstrate that learning the conformity-score geometry substantially improves prediction-region efficiency; specifically, *CoCoNet*$_{\text{ICNN}}$ achieves the top efficiency rank on 22 out of 30 tasks. By bridging expressive score optimization with strict regional convexity, *CoCoNet* provides a principled foundation for generating tight and tractable uncertainty sets for multi-dimensional regression settings. Future work should focus on learning efficient prediction regions while considering conditional and group-conditional coverage.

Code and Data

The code and datasets used in this paper are publicly available at `https://github.com/AI4OPT/CoCoNet`. All reported numerical results, tables, and figures can be reproduced using the provided code and data under the same experimental configuration and computational environment described in the paper.

Acknowledgment

This work was funded by NSF award 2112533.

References

Amos, B., Xu, L., and Kolter, J. Z. (2017). Input convex neural networks. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 146–155. PMLR.

Bai, Y., Mei, S., Wang, H., Zhou, Y., and Xiong, C. (2022). Efficient and differentiable conformal prediction with general function classes. In *International Conference on Learning Representations*.

Bellotti, A. (2020). Constructing normalized nonconformity measures based on maximizing predictive efficiency. In *Conformal and Probabilistic Prediction and Applications*, pages 41–54. PMLR.

Bertsimas, D., Brown, D. B., and Caramanis, C. (2011). Theory and applications of robust optimization. *SIAM review*, 53(3):464–501.

Blum, L. C. and Reymond, J.-L. (2009). 970 million druglike small molecules for virtual screening in the chemical universe database GDB-13. *Journal of the American Chemical Society*, 131(25):8732–8733.

Braun, S., Aolaritei, L., Jordan, M. I., and Bach, F. (2026). Minimum volume conformal sets for multivariate regression. *Journal of the American Statistical Association*, (just-accepted):1–27.

Carlier, G., Chernozhukov, V., and Galichon, A. (2016). Vector quantile regression: An optimal transport approach. *The Annals of Statistics*, 44(3):1165 – 1192.

Cevid, D., Michel, L., Näf, J., Bühlmann, P., and Meinshausen, N. (2022). Distributional random forests: Heterogeneity adjustment and multivariate distributional regression. *Journal of Machine Learning Research*, 23(333):1–79.

Chakraborty, B. (2003). On multivariate quantile regression. *Journal of statistical planning and inference*, 110(1-2):109–132.

Chen, D. (2012). Online retail ii. UCI Machine Learning Repository.

Chen, S., Zhou, W., and Zhu, S. (2026). Learning polyhedral conformal sets for robust optimization. *arXiv preprint arXiv:2605.08506*.

Chenreddy, A. R. and Delage, E. (2024). End-to-end conditional robust optimization. In Kiyavash, N. and Mooij, J. M., editors, *Proceedings of the Fortieth Conference on Uncertainty in Artificial Intelligence*, volume 244 of *Proceedings of Machine Learning Research*, pages 736–748. PMLR.

Cho, D., Yoo, C., Im, J., and Cha, D.-H. (2020). Comparative assessment of various machine learning-based bias correction methods for numerical weather prediction model forecasts of extreme air temperatures in urban areas.

Colombo, N. (2024). Normalizing flows for conformal regression. *arXiv preprint arXiv:2406.03346*.

Dheur, V., Fontana, M., Estievenart, Y., Desobry, N., and Taieb, S. B. (2025). A unified comparative study with generalized conformity scores for multi-output conformal regression. *arXiv preprint arXiv:2501.10533*.

English, E. and Lippert, C. (2026). Japan: Joint adaptive prediction areas with normalising flow. In Vondrick, C., Hariharan, B., Raffel, C., Pinto, L., Yang, D., and Faust, A., editors, *International Conference on Learning Representations*, volume 2026, pages 131010–131047.

Fanaee-T, H. (2013). Bike sharing. UCI Machine Learning Repository.

Fang, Z., Tan, A., and Huang, J. (2025). CONTRA: Conformal prediction region via normalizing flow transformation. In Yue, Y., Garg, A., Peng, N., Sha, F., and Yu, R., editors, *International Conference on Learning Representations*, volume 2025, pages 91833–91850.

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

Gao, C., Shan, L., Srinivas, V., and Vijayaraghavan, A. (2025). Volume optimality in conformal prediction with structured prediction sets. *arXiv preprint arXiv:2502.16658*.

Hallin, M., Paindaveine, D., Šiman, M., Wei, Y., Serfling, R., Zuo, Y., Kong, L., and Mizera, I. (2010). Multivariate quantiles and multiple-output regression quantiles: From l_1 optimization to halfspace depth [with discussion and rejoinder]. *The Annals of Statistics*, pages 635–703.

Hebrail, G. and Berard, A. (2006). Individual household electric power consumption. UCI Machine Learning Repository.

Johnstone, C. and Cox, B. (2021). Conformal uncertainty sets for robust optimization. In *Conformal and Probabilistic Prediction and Applications*.

Kondratyev, V., Fishkov, A., Kotelevskii, N., Hegazy, M., Flamary, R., Panov, M., and Moulines, E. (2025). Neural optimal transport meets multivariate conformal prediction. *arXiv preprint arXiv:2509.25444*.

Lavikainen, J., Stenroth, L., Alkjær, T., Karjalainen, P. A., Korhonen, R. K., and Mononen, M. E. (2023). Prediction of knee joint compartmental loading maxima utilizing simple subject characteristics and neural networks. *Annals of Biomedical Engineering*, 51(11):2479–2489.

Lavikainen, J. J. and Stenroth, L. (2022). Dataset of knee joint contact force peaks and corresponding subject characteristics from 4 open datasets. Version 1.0.0.

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

Liang, X., Zou, T., Guo, B., Li, S., Zhang, H., Zhang, S., Huang, H., and Chen, S. X. (2015). Assessing beijing's pm2.5 pollution: severity, weather impact, apec and winter heating. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 471(2182):20150257.

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

Messoudi, S., Destercke, S., and Rousseau, S. (2022). Ellipsoidal conformal inference for multi-target regression. In *Conformal and Probabilistic Prediction with Applications*, pages 294–306.

Miller, C., Kathirgamanathan, A., Picchetti, B., Arjunan, P., Park, J. Y., Nagy, Z., Raftery, P., Hobson, B. W., Shi, Z., and Meggers, F. (2020). The building data genome project 2, energy meter data from the ashrae great energy predictor iii competition. *Scientific Data*, 7(1):368.

Montavon, G., Rupp, M., Gobre, V., Vazquez-Mayagoitia, A., Hansen, K., Tkatchenko, A., Müller, K.-R., and von Lilienfeld, O. A. (2013). Machine learning of molecular electronic properties in chemical compound space. *New Journal of Physics*, 15(9):095003.

Paindaveine, D. and Šiman, M. (2011). On directional multiple-output quantile regression. *Journal of multivariate analysis*, 102(2):193–212.

Papadopoulos, H., Proedrou, K., Vovk, V., and Gammerman, A. (2002). Inductive confidence machines for regression. In *Machine Learning: ECML 2002*, pages 345–356. Springer.

Paredes, E. and Ballester-Ripoll, R. (2017). Sgemm gpu kernel performance. UCI Machine Learning Repository.

Plassier, V., Fishkov, A., Dheur, V., Guizani, M., Taieb, S. B., Panov, M., and Moulines, E. (2025). Rectifying conformity scores for better conditional coverage. *arXiv preprint arXiv:2502.16336*.

Ramakrishnan, R., Dral, P. O., Rupp, M., and von Lilienfeld, O. A. (2014). Quantum chemistry structures and properties of 134 kilo molecules. *Scientific Data*, 1:140022.

Rana, P. (2013). Physicochemical properties of protein tertiary structure. UCI Machine Learning Repository.

Rasmussen, C. E., Neal, R. M., Hinton, G. E., van Camp, D., Revow, M., Ghahramani, Z., Kustra, R., and Tibshirani, R. (1996). DELVE: Data for evaluating learning in valid experiments. The kin8nm dataset is part of the DELVE benchmark collection.

Redmond, M. (2009). Communities and crime unnormalized. UCI Machine Learning Repository.

Romano, Y., Patterson, E., and Candès, E. J. (2019). Conformalized quantile regression. In *Advances in Neural Information Processing Systems*, volume 32.

Sampson, M. and Chan, K.-S. (2024). Conformal multi-target hyperrectangles. *Statistical Analysis and Data Mining: The ASA Data Science Journal*, 17(5):e11710.

Schlembach, F., Smirnov, E., and Winands, M. H. (2025). Dynamic conformal prediction for multi-target regression: Optimising informational efficiency under joint validity. In *Fourteenth Symposium on Conformal and Probabilistic Prediction with Applications (COPA 2025)*, pages 193–213. PMLR.

Schneider, R. (2014). *Convex Bodies: The Brunn–Minkowski Theory*. Cambridge University Press, 2 edition.

Sharma, A., Veer, S., Hancock, A., Yang, H., Pavone, M., and Majumdar, A. (2023). Pac-bayes generalization certificates for learned inductive conformal prediction. *Advances in Neural Information Processing Systems*, 36:46807–46829.

Spyromitros-Xioufis, E., Tsoumakas, G., Groves, W., and Vlahavas, I. (2016). Multi-target regression via input space expansion: Treating targets as inputs. *Machine Learning*, 104(1):55–98.

Stutz, D., Dvijotham, K., Cemgil, A. T., and Doucet, A. (2022). Learning optimal conformal classifiers. In *International Conference on Learning Representations*.

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

Tsanas, A. and Xifara, A. (2012). Accurate quantitative estimation of energy performance of residential buildings using statistical machine learning tools. *Energy and Buildings*, 49:560–567.

Tüfekci, P. (2014). Prediction of full load electrical power output of a base load operated combined cycle power plant using machine learning methods. *International Journal of Electrical Power & Energy Systems*, 60:126–140.

Tumu, R., Cleaveland, M., Mangharam, R., Pappas, G., and Lindemann, L. (2024). Multi-modal conformal prediction regions by optimizing convex shape templates. In *6th Annual Learning for Dynamics & Control Conference*, pages 1343–1356. PMLR.

Vijayakumar, S. and Schaal, S. (2000). Fast and efficient incremental learning for high-dimensional movement systems. In *Proceedings of the IEEE International Conference on Robotics and Automation*, volume 2, pages 1894–1899.

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

Wang, I., Van Parys, B., and Stellato, B. (2023). Learning decision-focused uncertainty sets in robust optimization.

Wilson, B., Qi, W., Agarwal, T., Lambert, J., Singh, J., Khandelwal, S., Pan, B., Kumar, R., Hartnett, A., Pontes, J. K., Ramanan, D., Carr, P., and Hays, J. (2021). Argoverse 2: Next generation datasets for self-driving perception and forecasting. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*.

Wu, Z., Ramsundar, B., Feinberg, E. N., Gomes, J., Geniesse, C., Pappu, A. S., Leswing, K., and Pande, V. (2018). Moleculenet: A benchmark for molecular machine learning. *Chemical Science*, 9:513–530.

Yeh, C., Christianson, N., Wu, A., Wierman, A., and Yue, Y. (2025). End-to-end conformal calibration for optimization under uncertainty. *Transactions on Machine Learning Research*.

Yeh, I.-C. (1998). Concrete compressive strength. UCI Machine Learning Repository.

Zhang, S., Guo, B., Dong, A., He, J., Xu, Z., and Chen, S. X. (2017). Cautionary tales on air-quality improvement in beijing. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 473(2205):20170457.

Zhou, F., Claire, Q., and King, R. D. (2014). Predicting the geographical origin of music. In *2014 IEEE International Conference on Data Mining*.

A Related Work

Multivariate Uncertainty Quantification. Uncertainty quantification for multivariate responses has been studied through geometric and directional quantile methods Chakraborty (2003); Hallin et al. (2010); Paindaveine and Šiman (2011), vector quantile regression based on optimal transport Carlier et al. (2016), and distributional approaches that model the joint conditional distribution of the response Cevid et al. (2022). These approaches characterize joint uncertainty through multivariate quantiles, depth regions, or conditional distributional representations. Their theoretical guarantees are generally tied to the corresponding statistical formulation and regularity or specification assumptions, such as consistency or asymptotic validity, rather than finite-sample distribution-free predictive coverage. Conformal prediction addresses a complementary objective by directly calibrating prediction regions to provide finite-sample marginal coverage under exchangeability.

Structured Multivariate Conformal Prediction. Multivariate conformal methods differ primarily in the geometric structure imposed on the joint prediction region and in how cross-output dependence is represented. Copula-based approaches combine target-wise conformity information while explicitly modeling dependence across outputs Messoudi et al. (2021). Coordinatewise conformalization naturally yields axis-aligned hyperrectangles, with recent methods further optimizing their widths or dimension-wise error allocations to improve efficiency Sampson and Chan (2024); Schlembach et al. (2025). Covariance- and Mahalanobis-based conformity scores instead produce ellipsoidal regions that capture cross-output dependence and heterogeneous uncertainty Johnstone and Cox (2021); Messoudi et al. (2022). Tumu et al. (2024) optimize parameterized convex shape templates and combine multiple convex components to construct tractable multimodal conformal prediction regions for downstream planning and control. Minimum-Volume Conformal Sets (MVCS) directly optimize prediction-set size over adaptive single and multi-norm families Braun et al. (2026). Compared with basic coordinatewise constructions, these methods improve efficiency by modeling cross-output dependence and adapting the region geometry. However, convex approaches typically remain restricted to pre-specified families such as hyperrectangles, ellipsoids, or norm-based sets.

Flexible Multivariate Conformal Prediction. Recent multivariate conformal methods move beyond predefined geometric families by learning transformations or representations of the joint response distribution. Feldman et al. (2023) use representation learning to construct calibrated multiple-output prediction regions, while optimal-transport approaches use multivariate ranks and vector quantiles to obtain geometry-adaptive regions Thurin et al. (2025); Kondratyev et al. (2025). Normalizing-flow methods provide another route to geometric flexibility: CONTRA and its residual-based extension learn transformations between the output or residual space and a simpler latent space Fang et al. (2025), while JAPAN constructs adaptive regions by conformalizing flow-based density scores English and Lippert (2026). While these methods are not restricted to pre-defined geometries, they may construct highly sophisticated and non-convex prediction regions.

Decision-Focused Uncertainty-Set Learning. Motivated by the role of uncertainty sets in downstream optimization, Wang et al. (2023) learn contextual uncertainty sets by differentiating through robust optimization problems, while Chenreddy and Delage (2024) develop an end-to-end conditional robust optimization framework that jointly considers decision performance and uncertainty-set quality. More closely related to conformal uncertainty sets, Yeh et al. (2025) use partially input-convex neural networks to integrate conformal calibration with robust optimization, and Chen et al. (2026) optimize polyhedral conformal uncertainty sets directly for robust decision loss. These approaches tailor uncertainty-set learning to a specified downstream problem. While this task-specific design can improve decision performance, it couples the learned sets to the particular objective, constraints, and optimization structure. Consequently, a change in the downstream problem may require reformulation or retraining, and prediction-region efficiency is not itself the primary learning objective.

Conformity Score Learning and Prediction-Set Efficiency. Standard conformal regression commonly constructs conformity scores from absolute or normalized prediction residuals Lei et al. (2018). CQR adapts conformal intervals to heteroscedastic and asymmetric uncertainty through learned conditional quantiles Romano et al. (2019), while normalized nonconformity measures have been parameterized and optimized explicitly for predictive efficiency Bellotti (2020). More general approaches optimize conformal predictors or prediction-set families directly, including conformal training Stutz et al. (2022), CP-Gen Bai et al. (2022), and

joint learning of predictive models and conformity-score parameters Sharma et al. (2023). Other methods learn adaptive score transformations or richer scoring representations to improve conditional behavior and efficiency Colombo (2024); Plassier et al. (2025); Dheur et al. (2025). Recent work has also studied volume optimality relative to structured prediction-set classes Gao et al. (2025). These works demonstrate that conformity scores and prediction-set parameters can themselves be learned to improve adaptivity and efficiency. Building on this perspective, *CoCoNet* learns within an admissible class of convex conformity scores, enforcing convexity structurally while directly optimizing the size of the induced sublevel sets. This allows the prediction-region geometry to adapt to the input without restricting it to a predefined convex shape family.

B Theoretical Results and Proofs

B.1 Marginal Coverage

The conformity score is learned using D_{score} and is fixed before the calibration set is used. Therefore, conditional on the learned $S_{\boldsymbol{\theta}}$, standard split-conformal validity applies (Lei et al., 2018).

Let $\widehat{\tau}_\alpha$ be the conformal threshold defined in (3), and let $\mathcal{C}_{\boldsymbol{\theta}}(\mathbf{x})$ be the corresponding prediction set defined in (2).

Proposition 4 (Marginal coverage)**.** *Suppose that the calibration observations and a new test observation are exchangeable, and that $S_{\boldsymbol{\theta}}$ is independent of D_{cal}. Then*

$$\mathbb{P}\left(\mathbf{Y} \in \mathcal{C}_{\boldsymbol{\theta}}(\mathbf{X})\right) \geq 1 - \alpha.$$

If the calibration and test conformity scores are almost surely distinct, then

$$1 - \alpha \leq \mathbb{P}\left(\mathbf{Y} \in \mathcal{C}_{\boldsymbol{\theta}}(\mathbf{X})\right) \leq 1 - \alpha + \frac{1}{n_{\text{cal}} + 1}.$$

Proof. Let

$$\widehat{\tau}_\alpha = s_{(\lceil (n_{\text{cal}}+1)(1-\alpha) \rceil)}$$

be the conformal quantile of the calibration scores $s_i = S_{\boldsymbol{\theta}}(\mathbf{X}_i, \mathbf{Y}_i - F(\mathbf{X}_i))$.

Since $S_{\boldsymbol{\theta}}$ is fixed before calibration, exchangeability of the calibration and test observations implies exchangeability of their conformity scores. Therefore, the rank of the test score among the $n_{\text{cal}} + 1$ calibration and test scores is uniform.

Consequently, the probability that the test score does not exceed $\widehat{\tau}_\alpha$ is at least $1 - \alpha$. Since

$$\mathbf{Y} \in \mathcal{C}_{\boldsymbol{\theta}}(\mathbf{X}) \quad \Longleftrightarrow \quad S_{\boldsymbol{\theta}}(\mathbf{X}, \mathbf{Y} - F(\mathbf{X})) \leq \widehat{\tau}_\alpha,$$

it follows that

$$\mathbb{P}\left(\mathbf{Y} \in \mathcal{C}_{\boldsymbol{\theta}}(\mathbf{X})\right) \geq 1 - \alpha.$$

When the scores are almost surely distinct, the standard split-conformal rank argument further gives

$$\mathbb{P}\left(\mathbf{Y} \in \mathcal{C}_{\boldsymbol{\theta}}(\mathbf{X})\right) \leq 1 - \alpha + \frac{1}{n_{\text{cal}} + 1},$$

which establishes the result (Lei et al., 2018). □

B.2 Convexity and Point-Prediction Inclusion

Let $S \in \mathcal{S}_{\text{adm}}$, where $\mathcal{S}_{\text{adm}}$ is defined in (6). For any $\mathbf{x} \in \mathcal{X}$ and $\tau \geq 0$, define

$$\mathcal{C}(\mathbf{x}; S, \tau) = \left\{\mathbf{y} \in \mathcal{Y} : S\big(\mathbf{x}, \mathbf{y} - F(\mathbf{x})\big) \leq \tau\right\}.$$

Proposition 5. *For any $S \in \mathcal{S}_{adm}$, $\mathbf{x} \in \mathcal{X}$, and $\tau \geq 0$, the induced prediction set $\mathcal{C}(\mathbf{x}; S, \tau)$ is convex. Moreover,*

$$F(\mathbf{x}) \in \mathcal{C}(\mathbf{x}; S, \tau).$$

Proof. Fix $\mathbf{x} \in \mathcal{X}$ and let $\mathbf{y}_1, \mathbf{y}_2 \in \mathcal{C}(\mathbf{x}; S, \tau)$. By definition,

$$S\big(\mathbf{x}, \mathbf{y}_1 - F(\mathbf{x})\big) \le \tau, \qquad S\big(\mathbf{x}, \mathbf{y}_2 - F(\mathbf{x})\big) \le \tau.$$

For any $\lambda \in [0, 1]$, define

$$\mathbf{y}_\lambda = \lambda \mathbf{y}_1 + (1 - \lambda)\mathbf{y}_2.$$

Then

$$\mathbf{y}_\lambda - F(\mathbf{x}) = \lambda \mathbf{y}_1 + (1 - \lambda)\mathbf{y}_2 - F(\mathbf{x}) \tag{23}$$

$$= \lambda\big(\mathbf{y}_1 - F(\mathbf{x})\big) + (1 - \lambda)\big(\mathbf{y}_2 - F(\mathbf{x})\big). \tag{24}$$

Since $S(\mathbf{x}, \cdot)$ is convex,

$$S\big(\mathbf{x}, \mathbf{y}_\lambda - F(\mathbf{x})\big) \le \lambda S\big(\mathbf{x}, \mathbf{y}_1 - F(\mathbf{x})\big) + (1 - \lambda) S\big(\mathbf{x}, \mathbf{y}_2 - F(\mathbf{x})\big) \tag{25}$$

$$\le \lambda\tau + (1 - \lambda)\tau \tag{26}$$

$$= \tau. \tag{27}$$

Therefore,

$$\mathbf{y}_\lambda \in \mathcal{C}(\mathbf{x}; S, \tau),$$

which proves that $\mathcal{C}(\mathbf{x}; S, \tau)$ is convex.

Moreover, since $S(\mathbf{x}, 0) = 0$ and $\tau \ge 0$,

$$S\big(\mathbf{x}, F(\mathbf{x}) - F(\mathbf{x})\big) = S(\mathbf{x}, 0) = 0 \le \tau.$$

Hence,

$$F(\mathbf{x}) \in \mathcal{C}(\mathbf{x}; S, \tau).$$

Therefore, the induced prediction set is nonempty and contains the point prediction $F(\mathbf{x})$. □

B.3 Admissibility of the *CoCoNet* Score Parameterizations

We now verify that the two *CoCoNet* score parameterizations satisfy the admissibility conditions in (6). Recall that $S_{\boldsymbol{\theta}} \in \mathcal{S}_{\mathrm{adm}}$ if, for every $\mathbf{x} \in \mathcal{X}$, the function $S_{\boldsymbol{\theta}}(\mathbf{x}, \cdot)$ is convex, nonnegative, and satisfies $S_{\boldsymbol{\theta}}(\mathbf{x}, 0) = 0$.

CoCoNet$_{\mathrm{Spline}}$**.** Fix $\mathbf{x} \in \mathcal{X}$. For each coordinate j and branch $\pm$, the spline slopes $c_{j,\pm,k}$ are positive and nondecreasing by construction. Hence, each one-sided spline function $G_{j,\pm} : \mathbb{R}_+ \to \mathbb{R}_+$ is convex, nondecreasing, nonnegative, and satisfies $G_{j,\pm}(0) = 0$.

Define

$$g_j(t) = G_{j,+}([t]_+) + G_{j,-}([-t]_+),$$

where $[t]_+ = \max\{t, 0\}$. Since $[t]_+$ and $[-t]_+$ are convex, and $G_{j,+}$ and $G_{j,-}$ are convex and nondecreasing, g_j is convex and nonnegative, with $g_j(0) = 0$.

For fixed $\mathbf{x}$, the rotation matrix $Q = Q(\mathbf{x})$ is independent of $\mathbf{r}$. Writing $\mathbf{q}_j$ for its jth column,

$$z_j = \mathbf{q}_j^\top \mathbf{r}$$

is linear in $\mathbf{r}$. Therefore, $g_j(\mathbf{q}_j^\top \mathbf{r})$ is convex in $\mathbf{r}$. The spline score can be written as

$$S_{\boldsymbol{\theta}}^{\mathrm{Spline}}(\mathbf{x}, \mathbf{r}) = \max_{1 \le j \le p} g_j(\mathbf{q}_j^\top \mathbf{r}).$$

Since a pointwise maximum of convex functions is convex, $S_{\boldsymbol{\theta}}^{\mathrm{Spline}}(\mathbf{x}, \cdot)$ is convex. Moreover, each g_j is nonnegative and vanishes at zero, so

$$S_{\boldsymbol{\theta}}^{\mathrm{Spline}}(\mathbf{x}, \mathbf{r}) \ge 0, \qquad S_{\boldsymbol{\theta}}^{\mathrm{Spline}}(\mathbf{x}, 0) = 0.$$

Thus,

$$S_{\boldsymbol{\theta}}^{\mathrm{Spline}} \in \mathcal{S}_{\mathrm{adm}}.$$

The proof follows a similar procedure for the case with ℓ_1 aggregation. The score in this case is the sum of the same functions $g_j(\mathbf{q}_j^\top \mathbf{r})$; a sum of convex, nonnegative functions that vanish at zero is again convex, nonnegative, and vanishes at zero, so *CoCoNet*$_{\mathrm{Spline}\text{-}\ell_1}$ is admissible as well.

CoCoNet$_{\mathrm{ICNN}}$**.** Fix $\mathbf{x} \in \mathcal{X}$. By the ICNN construction, the nonnegativity constraints on the hidden-to-hidden weights together with the use of convex, nondecreasing activation functions ensure that

$$\mathbf{r} \mapsto G_{\boldsymbol{\theta}}(\mathbf{x}, \mathbf{r})$$

is convex (Amos et al., 2017).

By the first-order inequality for differentiable convex functions,

$$G_{\boldsymbol{\theta}}(\mathbf{x}, \mathbf{r}) \geq G_{\boldsymbol{\theta}}(\mathbf{x}, 0) + \nabla_{\mathbf{r}} G_{\boldsymbol{\theta}}(\mathbf{x}, 0)^{\top} \mathbf{r}.$$

Hence, the corresponding Bregman divergence

$$D_G(\mathbf{r}, 0) = G_{\boldsymbol{\theta}}(\mathbf{x}, \mathbf{r}) - G_{\boldsymbol{\theta}}(\mathbf{x}, 0) - \nabla_{\mathbf{r}} G_{\boldsymbol{\theta}}(\mathbf{x}, 0)^{\top} \mathbf{r}$$

is nonnegative and satisfies $D_G(0, 0) = 0$.

Moreover, $D_G(\cdot, 0)$ is convex because it is obtained from the convex function $G_{\boldsymbol{\theta}}(\mathbf{x}, \cdot)$ by subtracting an affine function. Therefore,

$$S_{\boldsymbol{\theta}}^{\mathrm{ICNN}}(\mathbf{x}, \mathbf{r}) = D_G(\mathbf{r}, 0) + \epsilon_{\mathrm{ICNN}} \|\mathbf{r}\|_2^2$$

is convex and nonnegative for $\epsilon_{\mathrm{ICNN}} > 0$. Evaluating at $\mathbf{r} = \mathbf{0}$ gives

$$S_{\boldsymbol{\theta}}^{\mathrm{ICNN}}(\mathbf{x}, 0) = 0.$$

Thus,

$$S_{\boldsymbol{\theta}}^{\mathrm{ICNN}} \in \mathcal{S}_{\mathrm{adm}}.$$

B.4 Convex-Score Representation

The following result establishes the converse relationship between convex prediction regions and the admissible score class.

Proposition 6 (Convex-score representation)**.** *Let $\mathcal{S}_{\mathrm{adm}}$ denote the admissible score class defined in* (6)*.*

For any $S \in \mathcal{S}_{\mathrm{adm}}$, $\mathbf{x} \in \mathcal{X}$, and $\tau \geq 0$, the induced prediction set

$$\mathcal{C}(\mathbf{x}; S, \tau) = \left\{ \mathbf{y} \in \mathbb{R}^p : S\big(\mathbf{x}, \mathbf{y} - F(\mathbf{x})\big) \leq \tau \right\}$$

is convex and contains $F(\mathbf{x})$.

Conversely, fix $\mathbf{x} \in \mathcal{X}$ and let $C(\mathbf{x}) \subset \mathbb{R}^p$ be a compact convex set satisfying

$$F(\mathbf{x}) \in \operatorname{int} C(\mathbf{x}).$$

Then there exists a nonnegative convex score

$$S_C(\mathbf{x}, \cdot) : \mathbb{R}^p \to [0, \infty)$$

satisfying

$$S_C(\mathbf{x}, 0) = 0$$

such that

$$C(\mathbf{x}) = \left\{ \mathbf{y} \in \mathbb{R}^p : S_C\big(\mathbf{x}, \mathbf{y} - F(\mathbf{x})\big) \leq 1 \right\}.$$

In particular, such a score is given by the Minkowski gauge of the translated set

$$K(\mathbf{x}) := C(\mathbf{x}) - F(\mathbf{x}),$$

namely

$$S_C(\mathbf{x}, \mathbf{r}) = \gamma_{K(\mathbf{x})}(\mathbf{r}) := \inf \left\{ t > 0 : \mathbf{r} \in tK(\mathbf{x}) \right\}.$$

Proof. The first direction follows from Proposition 5. It therefore remains to establish the converse.

Fix $\mathbf{x} \in \mathcal{X}$ and introduce the corresponding prediction region in residual coordinates,

$$K := C(\mathbf{x}) - F(\mathbf{x}) = \left\{ \mathbf{y} - F(\mathbf{x}) : \mathbf{y} \in C(\mathbf{x}) \right\}.$$

Translation preserves compactness and convexity. Moreover,

$$F(\mathbf{x}) \in \operatorname{int} C(\mathbf{x})$$

implies

$$0 \in \operatorname{int} K.$$

Define the Minkowski gauge of K by

$$\gamma_K(\mathbf{r}) = \inf \{t > 0 : \mathbf{r} \in tK\}.$$

It remains to verify that γ_K is finite and nonnegative, satisfies $\gamma_K(0) = 0$, is convex, and has K as its unit sublevel set.

Since $0 \in \operatorname{int} K$, there exists $\rho > 0$ such that the closed disk centered at then origin with radius ρ verifies

$$B_2(0, \rho) \subseteq K.$$

For an arbitrary $\mathbf{r} \in \mathbb{R}^p$, any

$$t \geq \frac{\|\mathbf{r}\|_2}{\rho}$$

satisfies

$$\frac{\mathbf{r}}{t} \in B_2(0, \rho) \subseteq K.$$

Consequently,

$$\mathbf{r} \in tK.$$

Thus, the set over which the infimum defining $\gamma_K(\mathbf{r})$ is taken is nonempty for every $\mathbf{r} \in \mathbb{R}^p$, and

$$0 \leq \gamma_K(\mathbf{r}) < \infty.$$

Since $0 \in tK$ for every $t > 0$,

$$\gamma_K(0) = 0.$$

Positive homogeneity follows directly from the definition. For any $c > 0$,

$$\begin{aligned} \gamma_K(c\mathbf{r}) &= \inf\{t > 0 : c\mathbf{r} \in tK\} \\ &= \inf\{cs : s > 0,\ \mathbf{r} \in sK\} \\ &= c\,\gamma_K(\mathbf{r}). \end{aligned}$$

For $c = 0$, the same identity follows from $\gamma_K(0) = 0$. Hence,

$$\gamma_K(c\mathbf{r}) = c\,\gamma_K(\mathbf{r}), \qquad c \geq 0.$$

Subadditivity follows from convexity of K. Let $\mathbf{r}_1, \mathbf{r}_2 \in \mathbb{R}^p$ and let $\varepsilon > 0$. By the definition of the infimum, there exist feasible $a, b > 0$ satisfying

$$\mathbf{r}_1 \in aK, \qquad a < \gamma_K(\mathbf{r}_1) + \varepsilon,$$

and

$$\mathbf{r}_2 \in bK, \qquad b < \gamma_K(\mathbf{r}_2) + \varepsilon.$$

Hence, for some $\mathbf{k}_1, \mathbf{k}_2 \in K$,

$$\mathbf{r}_1 = a\mathbf{k}_1, \qquad \mathbf{r}_2 = b\mathbf{k}_2.$$

It follows that

$$\mathbf{r}_1 + \mathbf{r}_2 = (a + b)\left(\frac{a}{a+b}\mathbf{k}_1 + \frac{b}{a+b}\mathbf{k}_2\right).$$

The coefficients in parentheses are nonnegative and sum to one. Convexity of K therefore implies

$$\frac{a}{a+b}\mathbf{k}_1 + \frac{b}{a+b}\mathbf{k}_2 \in K.$$

Consequently,

$$\mathbf{r}_1 + \mathbf{r}_2 \in (a+b)K,$$

and hence

$$\gamma_K(\mathbf{r}_1 + \mathbf{r}_2) \leq a + b < \gamma_K(\mathbf{r}_1) + \gamma_K(\mathbf{r}_2) + 2\varepsilon.$$

Since $\varepsilon > 0$ is arbitrary,

$$\gamma_K(\mathbf{r}_1 + \mathbf{r}_2) \leq \gamma_K(\mathbf{r}_1) + \gamma_K(\mathbf{r}_2).$$

Combining subadditivity with positive homogeneity yields convexity. For any $\lambda \in [0,1]$,

$$\begin{aligned}\gamma_K\big(\lambda\mathbf{r}_1 + (1-\lambda)\mathbf{r}_2\big) &\leq \gamma_K(\lambda\mathbf{r}_1) + \gamma_K((1-\lambda)\mathbf{r}_2) \\ &= \lambda\gamma_K(\mathbf{r}_1) + (1-\lambda)\gamma_K(\mathbf{r}_2).\end{aligned}$$

Thus γ_K is convex.

It remains to establish that K is exactly the unit sublevel set of γ_K.

If $\mathbf{r} \in K$, then $\mathbf{r} \in 1K$, so

$$\gamma_K(\mathbf{r}) \leq 1.$$

Therefore,

$$K \subseteq \{\mathbf{r} : \gamma_K(\mathbf{r}) \leq 1\}.$$

For the reverse inclusion, suppose that

$$\gamma_K(\mathbf{r}) \leq 1.$$

By the definition of the infimum, there exists a sequence $\{t_n\}_{n\geq 1}$ such that

$$\mathbf{r} \in t_n K, \qquad t_n \longrightarrow \gamma_K(\mathbf{r}).$$

If $\gamma_K(\mathbf{r}) < 1$, then $t_n \leq 1$ for all sufficiently large n. Since $0 \in K$ and K is convex,

$$tK \subseteq K, \qquad t \in [0,1].$$

Hence,

$$\mathbf{r} \in t_n K \subseteq K$$

for all sufficiently large n, and therefore $\mathbf{r} \in K$.

If $\gamma_K(\mathbf{r}) = 1$, write

$$\mathbf{r} = t_n \mathbf{k}_n, \qquad \mathbf{k}_n \in K.$$

Then

$$\mathbf{k}_n = \frac{\mathbf{r}}{t_n} \longrightarrow \mathbf{r}.$$

Since K is compact and therefore closed,

$$\mathbf{r} \in K.$$

Thus,

$$\{\mathbf{r} : \gamma_K(\mathbf{r}) \leq 1\} \subseteq K.$$

Combining the two inclusions gives

$$K = \{\mathbf{r} : \gamma_K(\mathbf{r}) \leq 1\}.$$

Finally, define

$$S_C(\mathbf{x}, \mathbf{r}) := \gamma_K(\mathbf{r}).$$

The preceding arguments establish that $S_C(\mathbf{x}, \cdot)$ is finite, nonnegative, and convex, and that

$$S_C(\mathbf{x}, 0) = 0.$$

Moreover,

$$\begin{aligned}\big\{\mathbf{y} : S_C\big(\mathbf{x}, \mathbf{y} - F(\mathbf{x})\big) \leq 1\big\} &= F(\mathbf{x}) + \{\mathbf{r} : \gamma_K(\mathbf{r}) \leq 1\} \\ &= F(\mathbf{x}) + K \\ &= C(\mathbf{x}).\end{aligned}$$

This establishes the converse and completes the proof. □

B.5 Boundedness and well-posedness of the prediction regions.

In addition to convexity and inclusion of the point prediction, the proposed score families yield bounded finite sublevel sets. This ensures that the prediction-region volumes and boundary distances used below are well defined.

For *CoCoNet*$_{\text{Spline-}\ell_\infty}$, recall that every one-sided spline branch is zero-anchored and has slope bounded below by $\varepsilon_{\text{slope}} > 0$. Hence, for every $s \geq 0$,

$$G_{j,\pm}(s) \geq \varepsilon_{\text{slope}} s.$$

Let

$$\mathbf{z} = Q_{\boldsymbol{\theta}}\big(\mathbf{h}_{\boldsymbol{\theta}}(\mathbf{x})\big)^\top \mathbf{r}.$$

Using the definition of the spline score and $[z_j]_+ + [-z_j]_+ = |z_j|$, we obtain

$$S_{\boldsymbol{\theta}}^{\text{Spline}}(\mathbf{x}, \mathbf{r}) = \max_{1 \leq j \leq p} \{G_{j,+}([z_j]_+) + G_{j,-}([-z_j]_+)\} \tag{28}$$

$$\geq \varepsilon_{\text{slope}} \max_{1 \leq j \leq p} |z_j| \tag{29}$$

$$= \varepsilon_{\text{slope}} \|\mathbf{z}\|_\infty. \tag{30}$$

Since $Q_{\boldsymbol{\theta}}(h_{\boldsymbol{\theta}}(\mathbf{x}))$ is orthogonal, $\|\mathbf{z}\|_2 = \|\mathbf{r}\|_2$, and therefore

$$\|\mathbf{z}\|_\infty \geq \frac{\|\boldsymbol{z}\|_2}{\sqrt{p}} = \frac{\|\mathbf{r}\|_2}{\sqrt{p}}.$$

Consequently,

$$S_{\boldsymbol{\theta}}^{\text{Spline}}(\mathbf{x}, \mathbf{r}) \geq \frac{\varepsilon_{\text{slope}}}{\sqrt{p}} \|\mathbf{r}\|_2.$$

Thus, for every finite threshold τ,

$$S_{\boldsymbol{\theta}}^{\text{Spline}}(\mathbf{x}, \mathbf{r}) \leq \tau \quad \Longrightarrow \quad \|\mathbf{r}\|_2 \leq \frac{\sqrt{p}\,\tau}{\varepsilon_{\text{slope}}},$$

so every finite spline sublevel set is bounded. Since the spline score is continuous, these sublevel sets are also closed and hence compact. Moreover, the strictly positive branch slopes imply that each positive and negative coordinate boundary is uniquely defined. The proof for *CoCoNet*$_{\text{Spline-}\ell_1}$ follows a similar procedure.

For *CoCoNet*$_{\text{ICNN}}$, convexity of $G_{\boldsymbol{\theta}}(\mathbf{x}, \cdot)$ gives the first-order inequality

$$G_{\boldsymbol{\theta}}(\mathbf{x}, \mathbf{r}) \geq G_{\boldsymbol{\theta}}(\mathbf{x}, 0) + \nabla_{\mathbf{r}} G_{\boldsymbol{\theta}}(\mathbf{x}, 0)^\top \mathbf{r}.$$

Hence, from the definition of the ICNN score,

$$S_{\boldsymbol{\theta}}^{\text{ICNN}}(\mathbf{x}, \mathbf{r}) = G_{\boldsymbol{\theta}}(\mathbf{x}, \mathbf{r}) - G_{\boldsymbol{\theta}}(\mathbf{x}, 0) - \nabla_{\mathbf{r}} G_{\boldsymbol{\theta}}(\mathbf{x}, 0)^\top \mathbf{r} + \varepsilon_{\text{ICNN}} \|\mathbf{r}\|_2^2 \tag{31}$$

$$\geq \varepsilon_{\text{ICNN}} \|\mathbf{r}\|_2^2. \tag{32}$$

Therefore,

$$S_{\boldsymbol{\theta}}^{\text{ICNN}}(\mathbf{x}, \mathbf{r}) \leq \tau \quad \Longrightarrow \quad \|\mathbf{r}\|_2 \leq \sqrt{\frac{\tau}{\varepsilon_{\text{ICNN}}}},$$

so every finite ICNN sublevel set is bounded and, by continuity, compact. The quadratic term also makes $S_{\boldsymbol{\theta}}^{\text{ICNN}}(\mathbf{x}, \cdot)$ strongly convex, with its unique minimum at $\mathbf{r} = \mathbf{0}$. Fix a unit direction $\mathbf{u} \in \mathbb{S}^{p-1}$ and define

$$\psi_{\mathbf{u}}(\rho) = S_{\boldsymbol{\theta}}^{\text{ICNN}}(\mathbf{x}, \rho\mathbf{u}), \qquad \rho \geq 0.$$

Then $\psi_{\mathbf{u}}(0) = 0$, $\psi_{\mathbf{u}}$ is strictly increasing for $\rho > 0$, and

$$\psi_{\mathbf{u}}(\rho) \geq \varepsilon_{\text{ICNN}} \rho^2 \longrightarrow \infty \qquad \text{as } \rho \to \infty.$$

Consequently, for every $\tau > 0$ and every $\mathbf{u} \in \mathbb{S}^{p-1}$, there exists a unique finite radius $\rho_{\boldsymbol{\theta}}(\mathbf{x}, \mathbf{u}; \tau) > 0$ satisfying

$$S_{\boldsymbol{\theta}}^{\text{ICNN}}\big(\mathbf{x}, \rho_{\boldsymbol{\theta}}(\mathbf{x}, \mathbf{u}; \tau)\mathbf{u}\big) = \tau.$$

Thus, the coordinatewise spline boundaries and the radial ICNN boundaries used in the volume computations below are well defined.

C Optimization and Differentiation Details

C.1 Differentiability of the Score-Learning Objective

The score parameters are optimized using the objective in (9),

$$\mathcal{L}(\boldsymbol{\theta}) = \mathcal{L}_{\text{size}}(\boldsymbol{\theta}; \tau_{\text{tr}}) + \lambda_{\text{miss}} \mathcal{L}_{\text{miss}}(\boldsymbol{\theta}; \tau_{\text{tr}}) + \lambda_{\text{cov}} \mathcal{L}_{\text{cov}}(\boldsymbol{\theta}; \tau_{\text{tr}}). \tag{33}$$

This section describes how gradients of all three terms are obtained. Both score architectures are differentiable almost everywhere with respect to their parameters. The spline score, ReLU feature encoder, and ReLU miss penalty introduce piecewise-differentiable operations. Hence, standard subgradients are used at the remaining kink points.

For a mini-batch $\mathcal{B} = \{(\mathbf{x}_i, \mathbf{r}_i)\}_{i=1}^{|\mathcal{B}|}$, let

$$s_i(\boldsymbol{\theta}) = S_{\boldsymbol{\theta}}(\mathbf{x}_i, \mathbf{r}_i). \tag{34}$$

Miss penalty. The miss term is

$$\mathcal{L}_{\text{miss}} = \frac{1}{|\mathcal{B}|} \sum_{i=1}^{|\mathcal{B}|} \text{ReLU}\left(\frac{s_i(\boldsymbol{\theta})}{\tau_{\text{tr}}} - 1 \right). \tag{35}$$

Away from $s_i(\boldsymbol{\theta}) = \tau_{\text{tr}}$, its gradient is

$$\nabla_{\boldsymbol{\theta}} \mathcal{L}_{\text{miss}} = \frac{1}{|\mathcal{B}|\, \tau_{\text{tr}}} \sum_{i=1}^{|\mathcal{B}|} \mathbb{I}\left\{ s_i(\boldsymbol{\theta}) > \tau_{\text{tr}} \right\} \nabla_{\boldsymbol{\theta}} s_i(\boldsymbol{\theta}). \tag{36}$$

At $s_i(\boldsymbol{\theta}) = \tau_{\text{tr}}$, ReLU is nondifferentiable and a subgradient is used.

Coverage penalty. Define the smoothed inclusion indicator

$$c_i(\boldsymbol{\theta}) = \sigma\left(\frac{\tau_{\text{tr}} - s_i(\boldsymbol{\theta})}{T} \right), \qquad \widehat{\text{Cov}}_{\boldsymbol{\theta}} = \frac{1}{|\mathcal{B}|} \sum_{i=1}^{|\mathcal{B}|} c_i(\boldsymbol{\theta}). \tag{37}$$

Since the sigmoid is smooth,

$$\nabla_{\boldsymbol{\theta}} \widehat{\text{Cov}}_{\boldsymbol{\theta}} = -\frac{1}{|\mathcal{B}| T} \sum_{i=1}^{|\mathcal{B}|} c_i(\boldsymbol{\theta}) \big(1 - c_i(\boldsymbol{\theta})\big) \nabla_{\boldsymbol{\theta}} s_i(\boldsymbol{\theta}). \tag{38}$$

The coverage loss

$$\mathcal{L}_{\text{cov}} = \left[\widehat{\text{Cov}}_{\boldsymbol{\theta}} - (1 - \alpha) \right]^2 \tag{39}$$

therefore has gradient

$$\nabla_{\boldsymbol{\theta}} \mathcal{L}_{\text{cov}} = 2 \left[\widehat{\text{Cov}}_{\boldsymbol{\theta}} - (1 - \alpha) \right] \nabla_{\boldsymbol{\theta}} \widehat{\text{Cov}}_{\boldsymbol{\theta}}. \tag{40}$$

Prediction-region size. Let

$$V_{\boldsymbol{\theta}}(\mathbf{x}; \tau) = |\mathcal{C}_{\boldsymbol{\theta}}(\mathbf{x}; \tau)| \tag{41}$$

denote the p-dimensional volume of the prediction region.

Since translating the residual region by $F(\mathbf{x})$ does not change its volume, $V_{\boldsymbol{\theta}}$ can be computed directly in residual space. The size term is

$$\mathcal{L}_{\text{size}} = \left[\frac{1}{|\mathcal{B}|} \sum_{i=1}^{|\mathcal{B}|} V_{\boldsymbol{\theta}}(\mathbf{x}_i; \tau_{\text{tr}}) \right]^{1/p}. \tag{42}$$

Let

$$\overline{V}_{\boldsymbol{\theta}} = \frac{1}{|\mathcal{B}|} \sum_{i=1}^{|\mathcal{B}|} V_{\boldsymbol{\theta}}(\mathbf{x}_i; \tau_{\text{tr}}). \tag{43}$$

For $\overline{V}_{\boldsymbol{\theta}} > 0$,

$$\nabla_{\boldsymbol{\theta}} \mathcal{L}_{\text{size}} = \frac{1}{p} \overline{V}_{\boldsymbol{\theta}}^{1/p-1} \frac{1}{|\mathcal{B}|} \sum_{i=1}^{|\mathcal{B}|} \nabla_{\boldsymbol{\theta}} V_{\boldsymbol{\theta}}(\mathbf{x}_i; \tau_{\text{tr}}). \tag{44}$$

It therefore remains to show how $V_{\boldsymbol{\theta}}(\mathbf{x}; \tau)$ and its gradient are obtained for each proposed score family.

C.2 Spline Prediction-Region Volume and Differentiation

For *CoCoNet*$_{\text{Spline}}$, recall that the residual is represented in the rotated coordinates

$$\mathbf{z} = Q_{\boldsymbol{\theta}}\big(h_{\boldsymbol{\theta}}(\mathbf{x})\big)^{\top} \mathbf{r}. \tag{45}$$

At a fixed threshold τ, the corresponding sublevel set is an asymmetric axis-aligned hyperrectangle in $\mathbf{z}$-space. The strictly positive spline slopes ensure that each coordinate boundary is finite and uniquely defined.

Let $\rho_{j,+}(\mathbf{x}; \tau)$ and $\rho_{j,-}(\mathbf{x}; \tau)$ denote the positive and negative boundary distances along coordinate j. They are the unique solutions of

$$G_{j,+}\big(\rho_{j,+}(\mathbf{x}; \tau); h_{\boldsymbol{\theta}}(\mathbf{x}), \boldsymbol{\theta}\big) = \tau, \tag{46}$$

$$G_{j,-}\big(\rho_{j,-}(\mathbf{x}; \tau); h_{\boldsymbol{\theta}}(\mathbf{x}), \boldsymbol{\theta}\big) = \tau. \tag{47}$$

Therefore, the width of the region along rotated coordinate j is $\rho_{j,+}(\mathbf{x}; \tau) + \rho_{j,-}(\mathbf{x}; \tau)$. Since $Q_{\boldsymbol{\theta}}(h_{\boldsymbol{\theta}}(\mathbf{x}))$ is orthogonal, the transformation from $\mathbf{r}$ to $\mathbf{z}$ preserves p-dimensional volume, and hence

$$V_{\boldsymbol{\theta}}(\mathbf{x}; \tau) = \prod_{j=1}^{p} \left(\rho_{j,+}(\mathbf{x}; \tau) + \rho_{j,-}(\mathbf{x}; \tau) \right). \tag{48}$$

Thus, for the spline parameterization, the prediction-region volume is available exactly from the coordinatewise boundary distances.

Differentiation through the spline boundaries. Within an active spline segment, each boundary radius is differentiable with respect to the score parameters. Implicitly differentiating (47) gives

$$\nabla_{\boldsymbol{\theta}} \rho_{j,\pm}(\mathbf{x}; \tau) = -\frac{\nabla_{\boldsymbol{\theta}} G_{j,\pm}\big(\rho_{j,\pm}(\mathbf{x}; \tau); h_{\boldsymbol{\theta}}(\mathbf{x}), \boldsymbol{\theta}\big)}{\partial_{\rho} G_{j,\pm}\big(\rho_{j,\pm}(\mathbf{x}; \tau); h_{\boldsymbol{\theta}}(\mathbf{x}), \boldsymbol{\theta}\big)}. \tag{49}$$

The numerator differentiates the spline branch with respect to $\boldsymbol{\theta}$ while holding the boundary radius fixed, and therefore includes the dependence through the feature encoder. The denominator is the slope of the active spline segment and is strictly positive by construction. At spline knots, the score is piecewise differentiable and the corresponding subgradient is used.

Differentiating (48) yields

$$\nabla_{\boldsymbol{\theta}} V_{\boldsymbol{\theta}}(\mathbf{x}; \tau) = V_{\boldsymbol{\theta}}(\mathbf{x}; \tau) \sum_{j=1}^{p} \frac{\nabla_{\boldsymbol{\theta}} \rho_{j,+}(\mathbf{x}; \tau) + \nabla_{\boldsymbol{\theta}} \rho_{j,-}(\mathbf{x}; \tau)}{\rho_{j,+}(\mathbf{x}; \tau) + \rho_{j,-}(\mathbf{x}; \tau)}. \tag{50}$$

Consequently, the exact spline-region volume is differentiable almost everywhere with respect to the score parameters and can be differentiated during optimization through its implicitly defined coordinate boundaries.

The ℓ_1 variant *CoCoNet*$_{\text{Spline-}\ell_1}$ uses the same rotation and one-sided spline branches but sums the coordinate scores instead of taking their maximum, so its prediction region is an asymmetric polytope rather than a hyperrectangle. Its volume is also available exactly: after the same change of coordinates, it reduces to an integral over a simplex of the inverse segment slopes, which evaluates to a finite sum of terms depending only on the threshold, the knot positions, and the slopes. This expression depends smoothly on the segment slopes, so the ℓ_1 volume is differentiable with respect to the score parameters and can be differentiated directly during optimization.

C.3 ICNN Prediction-Region Volume and Differentiation

Unlike *CoCoNet*$_{\text{Spline}}$, the *CoCoNet*$_{\text{ICNN}}$ region is generally not coordinatewise separable. Its volume is therefore computed from its radial boundary. Fix $\mathbf{x}$ and $\tau > 0$, and define the residual-space sublevel set

$$K_{\boldsymbol{\theta}}(\mathbf{x};\tau) = \left\{\mathbf{r} : S_{\boldsymbol{\theta}}^{\text{ICNN}}(\mathbf{x},\mathbf{r}) \le \tau\right\}. \tag{51}$$

Translation by $F(\mathbf{x})$ preserves volume, so $V_{\boldsymbol{\theta}}(\mathbf{x};\tau) = |K_{\boldsymbol{\theta}}(\mathbf{x};\tau)|$.

Strong convexity of $S_{\boldsymbol{\theta}}^{\text{ICNN}}(\mathbf{x},\cdot)$, together with $S_{\boldsymbol{\theta}}^{\text{ICNN}}(\mathbf{x},0) = 0$, implies that for every direction $\mathbf{u} \in \mathbb{S}^{p-1}$ there is a unique positive radius $\rho_{\boldsymbol{\theta}}(\mathbf{x},\mathbf{u};\tau)$ satisfying

$$S_{\boldsymbol{\theta}}^{\text{ICNN}}\big(\mathbf{x}, \rho_{\boldsymbol{\theta}}(\mathbf{x},\mathbf{u};\tau)\mathbf{u}\big) = \tau. \tag{52}$$

As proven formally in Appendix B.5. Using polar coordinates, the volume of the residual-space sublevel set can be written in terms of its radial boundary as described in Schneider (2014)

$$V_{\boldsymbol{\theta}}(\mathbf{x};\tau) = \frac{1}{p}\int_{\mathbb{S}^{p-1}} \rho_{\boldsymbol{\theta}}(\mathbf{x},\mathbf{u};\tau)^p \, d\sigma(\mathbf{u}), \tag{53}$$

where $d\sigma(\mathbf{u})$ denotes surface measure on the unit sphere $\mathbb{S}^{p-1}$. If $\mathbf{U} \sim \text{Unif}(\mathbb{S}^{p-1})$, then

$$\mathbb{E}[f(\mathbf{U})] = \frac{1}{|\mathbb{S}^{p-1}|}\int_{\mathbb{S}^{p-1}} f(\mathbf{u})\, d\sigma(\mathbf{u}). \tag{54}$$

Using $|\mathbb{S}^{p-1}| = p\, v_p$, where

$$v_p = \frac{\pi^{p/2}}{\Gamma(p/2+1)} \tag{55}$$

is the volume of the p-dimensional Euclidean unit ball, the radial volume identity can therefore be rewritten as

$$V_{\boldsymbol{\theta}}(\mathbf{x};\tau) = v_p\, \mathbb{E}_{\mathbf{U}\sim\text{Unif}(\mathbb{S}^{p-1})}\left[\rho_{\boldsymbol{\theta}}(\mathbf{x},\mathbf{U};\tau)^p\right]. \tag{56}$$

Given directions $\mathbf{u}_1,\ldots,\mathbf{u}_M$ generated by randomized quasi-Monte Carlo with uniform spherical marginals, the corresponding finite-direction estimator is

$$\widehat{V}_{\boldsymbol{\theta},M}(\mathbf{x};\tau) = \frac{v_p}{M}\sum_{m=1}^{M} \rho_{\boldsymbol{\theta}}(\mathbf{x},\mathbf{u}_m;\tau)^p. \tag{57}$$

Current implementation generates the directions using scrambled Sobol points.

Differentiation through the radial boundary. For fixed τ and fixed direction $\mathbf{u}$, (52) implicitly defines the boundary radius. Implicit differentiation gives

$$\nabla_{\boldsymbol{\theta}}\rho_{\boldsymbol{\theta}}(\mathbf{x},\mathbf{u};\tau) = -\frac{\nabla_{\boldsymbol{\theta}} S_{\boldsymbol{\theta}}^{\text{ICNN}}\big(\mathbf{x},\rho_{\boldsymbol{\theta}}(\mathbf{x},\mathbf{u};\tau)\mathbf{u}\big)}{\nabla_{\mathbf{r}} S_{\boldsymbol{\theta}}^{\text{ICNN}}\big(\mathbf{x},\rho_{\boldsymbol{\theta}}(\mathbf{x},\mathbf{u};\tau)\mathbf{u}\big)^{\top}\mathbf{u}}. \tag{58}$$

The numerator differentiates the score with respect to $\boldsymbol{\theta}$ while holding the boundary radius fixed and therefore includes the dependence through the feature encoder. The denominator is the directional derivative of the score along the ray. Strong convexity implies that this derivative is strictly positive at every positive-radius boundary.

For fixed sampled directions, differentiating (57) therefore gives

$$\nabla_{\boldsymbol{\theta}} \widehat{V}_{\boldsymbol{\theta},M}(\mathbf{x};\tau) = \frac{pv_p}{M} \sum_{m=1}^{M} \rho_{\boldsymbol{\theta}}(\mathbf{x},\mathbf{u}_m;\tau)^{p-1} \nabla_{\boldsymbol{\theta}} \rho_{\boldsymbol{\theta}}(\mathbf{x},\mathbf{u}_m;\tau). \tag{59}$$

Hence, although the ICNN boundary is obtained numerically, its volume estimator can be differentiated directly through the implicitly defined radial roots.

D Experimental Details and Configuration

Benchmark Datasets and Task Construction. Table 3 summarizes the datasets used in this paper, including 30 multi-dimensional datasets with response dimensions p ranging from 2 to 16, as well as six univariate datasets. The datasets cover a broad range of application domains, including energy, autonomous driving, environmental modeling, computer systems, robotics, and retail. This diversity provides a basis for evaluating the efficiency of *CoCoNet* across different application areas.

QM9. Four nested variants of this dataset are used, with response dimensions $p \in \{2, 4, 6, 8\}$. The number of observations and input features d_x are kept fixed across all variants. The response dimension is varied by selecting the first p molecular properties in a fixed ordering. All remaining molecular properties are excluded from the predictor set in every variant to avoid target leakage.

Argoverse2. This dataset is formulated as a multi-horizon trajectory-regression problem, with response dimensions $p \in \{2, 4, 8\}$. For each focal agent, five seconds of observed motion are used to predict future two-dimensional displacements relative to the agent's final observed position. The $p = 2$ task predicts the displacement at one future horizon, while the $p = 4$ and $p = 8$ tasks jointly predict displacements at additional future horizons. Therefore, increasing p changes the number of future positions predicted jointly, while the observations and input information remain unchanged. All predictors are constructed from the observed trajectory, and the targets are kept in the global city coordinate frame.

Argoverse 2 Motion. This dataset is used as a separate two-dimensional task with $p = 2$, where the response variables are `future_dx` and `future_dy`. For each scenario, one observation is constructed for the focal agent using the end of the five-second observed trajectory as the history anchor. No sliding-window augmentation is used. Although this task is derived from the same Argoverse 2 source data, it is different from Argoverse2-2D, which belongs to the nested multi-horizon variants described above.

Beijing Air Quality. Two multivariate tasks are constructed from this dataset, with $p = 6$ and $p = 12$. For the $p = 6$ task, each observation represents one station at one hour, and the responses are the concentrations of PM2.5, PM10, SO2, NO2, CO, and O3. Meteorological measurements, station identity, and calendar variables are used as predictors, resulting in $d_x = 37$ features after encoding. For the $p = 12$ task, each observation represents one hour, and the responses are the PM2.5 measurements from the 12 monitoring stations. Meteorological measurements from all stations and calendar variables are used as predictors, resulting in $d_x = 256$ features after encoding. Only hours with available PM2.5 measurements at all 12 stations are retained. Therefore, the $p = 6$ task represents dependence among pollutants within a station, while the $p = 12$ task represents spatial dependence across stations.

The remaining datasets follow their standard benchmark formulations; their domains, sample sizes, input dimensions, response dimensions, and source references are summarized in Table 3.

Synthetic Benchmark Construction. The synthetic benchmark is constructed so that the population-optimal convex prediction region at a prescribed coverage level $1 - \alpha$ is known. This allows the geometry and size of the learned regions to be evaluated against a known population reference. Four two-dimensional residual geometries are considered: an ellipsoid, rectangle, triangle, and trapezoid. In each setting, the response is generated according to:

Table 3: Datasets used in the reported experiments. n denotes the number of observations in the source dataset, d_x the number of input features, and p the response dimension. Multiple values of p denote tasks constructed at different response dimensions from the same source dataset.

Dataset	**Domain**	n	d_x	p
Univariate benchmarks				
Beijing PM2.5 (Liang et al., 2015)	Air quality	41,757	14	1
Bike Sharing (Fanaee-T, 2013)	Transportation	17,379	60	1
CCPP (Tüfekci, 2014)	Energy	9,568	4	1
Concrete (Yeh, 1998)	Civil engineering	1,030	8	1
Kin8nm (Rasmussen et al., 1996)	Robotics	8,192	8	1
Protein (Rana, 2013)	Structural biology	45,730	9	1
Multivariate benchmarks				
Argoverse 2 (Wilson et al., 2021)	Autonomous driving	24,988	28	2,4,8
Argoverse 2 Motion (Wilson et al., 2021)	Autonomous driving	24,988	28	2
Bias-Correction Temperature (Cho et al., 2020)	Meteorology	7,723	46	2
ENB (Tsanas and Xifara, 2012)	Building energy	768	8	2
Geographical Music (Zhou et al., 2014)	Music geolocation	1,059	68	2
CASP (Rana, 2013)	Structural biology	45,730	8	2
Online Retail II (Chen, 2012)	Retail	1,067,371	42	2
QM9 (Ramakrishnan et al., 2014)	Quantum chemistry	130,831	22	2,4,6,8
Household Power (Hebrail and Berard, 2006)	Energy	2,049,280	4	3
SCPF (Spyromitros-Xioufis et al., 2016)	Civic services	1,137	23	3
QM8 (Wu et al., 2018)	Quantum chemistry	21,786	14	4
SGEMM (Paredes and Ballester-Ripoll, 2017)	Computer systems	241,600	18	4
Beijing AQ (Zhang et al., 2017)	Air quality	383,980 / 28,663	37 / 256	6 / 12
Communities and Crime (Redmond, 2009)	Socioeconomic / crime	1,990	124	6
ATP1D (Spyromitros-Xioufis et al., 2016)	Airline ticket pricing	337	370	6
SARCOS (Vijayakumar and Schaal, 2000)	Robotics	48,933	21	7
Knee Contact Force (Lavikainen et al., 2023; Lavikainen and Stenroth, 2022)	Biomechanics	2,786	12	12
QM7b (Montavon et al., 2013; Blum and Reymond, 2009)	Quantum chemistry	7,211	29	14
WQ (Spyromitros-Xioufis et al., 2016)	Water quality	1,060	16	14
OES10 (Spyromitros-Xioufis et al., 2016)	Labor statistics	403	298	16
OES97 (Spyromitros-Xioufis et al., 2016)	Labor statistics	334	263	16
Building Data Genome 2, sites Fox / Eagle / Lamb (Miller et al., 2020)	Building energy	17,537 / 17,105 / 17,544	12	16

Large datasets are randomly subsampled to $20{,}000$ *observations, with the same sampled data used across all compared methods;* d_x *is the number of input features after categorical encoding of the data used in the experiments.*

$$\mathbf{x} \sim \mathrm{Unif}([-1,1]^2), \tag{60}$$

$$\mathbf{y} = \mu(\mathbf{x}) + \mathbf{r}, \tag{61}$$

where $\mu(\mathbf{x})$ is a known nonlinear conditional mean function. The exact conditional mean is used as the common point predictor for all methods so that the comparison focuses on prediction-region geometry rather than point-prediction error.

For each geometry, let $B \subset \mathbb{R}^2$ denote a convex body containing the origin in its interior. A latent residual $\mathbf{u}$ is generated by placing probability mass $1-\alpha$ uniformly inside B and the remaining probability mass uniformly in a larger, geometrically similar shell:

$$\mathbf{u} \sim \begin{cases} \mathrm{Unif}(B), & \text{with probability } 1-\alpha, \\ \mathrm{Unif}(cB \setminus B), & \text{with probability } \alpha, \end{cases} \qquad c > 1. \tag{62}$$

Table 4: Shared data-partitioning, preprocessing, and point-predictor configuration.

Component	**Configuration**
Random seeds	`{0,1,2,3,4}`
Data partition	`25/30/25/20%`
$\mathcal{D}_{\text{pred}}$ internal split	`85/15% train/validation`
$\mathcal{D}_{\text{score}}$ internal split	`80/20% train/validation`
Point predictor	`MLP`
Hidden dimensions	`[64,32]`
Activation	`ReLU`
Loss	`MSE`
Optimizer	`Adam`
Learning rate	`1e-3`
Batch size	`256`
Maximum epochs	`200`
Early stopping	`patience=20, min_delta=1e-5`

The residual is then transformed as

$$\mathbf{r} = \mathbf{A}\mathbf{u}, \tag{63}$$

$$\mathbf{A} = \mathbf{R}(\theta)\,\mathrm{diag}(a_1, a_2), \tag{64}$$

where $\mathbf{R}(\theta)$ represents a rotation and $a_1, a_2 > 0$ control scaling along the two coordinate directions. Hence, the inner region of the residual distribution is $\mathbf{A}B$, while the remaining probability mass is distributed over the surrounding shell $\mathbf{A}(cB \setminus B)$. Different choices of B and $\mathbf{A}$ produce residual distributions with different convex shapes, orientations, and aspect ratios under the same data-generating mechanism.

The expansion factor c is chosen so that the probability density inside the inner body is strictly larger than that in the surrounding shell. In particular, the two densities are

$$f_{\text{in}} = \frac{1-\alpha}{|B|}, \tag{65}$$

$$f_{\text{out}} = \frac{\alpha}{(c^2-1)|B|}, \tag{66}$$

with $f_{\text{in}} > f_{\text{out}}$. Since B contains exactly $1-\alpha$ probability mass and has strictly higher density than the surrounding shell, it is the population minimum-area region at the target coverage level. After the affine transformation, $\mathbf{A}B$ is the corresponding population minimum-area convex prediction region. Its area is available analytically as:

$$V^\star = |\det(\mathbf{A})|\,|B|. \tag{67}$$

This known oracle region provides the reference for evaluating the learned prediction regions using geometric overlap and relative volume.

Data Partitioning, Preprocessing, and Point Predictor. For each dataset and random seed, the observations are partitioned into four disjoint subsets, $\mathcal{D}_{\text{pred}}$, $\mathcal{D}_{\text{score}}$, $\mathcal{D}_{\text{cal}}$, and $\mathcal{D}_{\text{test}}$. The same partition is used across all compared methods for a given seed. $\mathcal{D}_{\text{pred}}$ is used exclusively to fit the point predictor, $\mathcal{D}_{\text{score}}$ is used to train and select the conformity-score model, $\mathcal{D}_{\text{cal}}$ is reserved for split-conformal calibration, and $\mathcal{D}_{\text{test}}$ is used only for final evaluation. Internal training and validation splits are used within $\mathcal{D}_{\text{pred}}$ and $\mathcal{D}_{\text{score}}$ for model fitting and selection, respectively.

Categorical features are one-hot encoded, while continuous features and response coordinates are standardized using statistics estimated exclusively from $\mathcal{D}_{\text{pred}}$. The same encoding and standardization parameters are then used for $\mathcal{D}_{\text{score}}$, $\mathcal{D}_{\text{cal}}$, and $\mathcal{D}_{\text{test}}$, so no preprocessing statistics are estimated from the calibration or test sets.

Table 5: Training configuration and hyperparameter search spaces for the proposed *CoCoNet* score families. The context encoder $\mathbf{h}(\mathbf{x})$ and loss-weight search spaces are shared across both families. Hyperparameter search is performed using only the validation portion of $\mathcal{D}_{\text{score}}$; values in braces are searched, while single values are fixed.

Component	**Hyperparameter**	**Value / Search Space**
Shared	Coverage temperature T	0.1
	Optimizer	`Adam`
	Gradient clipping	`5.0`
	λ_{cov}	`{0, 0.1, 1}`
	λ_{miss}	`{0.01, 0.03, 1, 10}`
	Training threshold τ_{tr}	1
Encoder $\mathbf{h}(\mathbf{x})$	Input	$[x, F(x)]$
	Hidden dimensions	`{[64,32], [256,128]}`
	Activation	`ReLU`
CoCoNet$_{\text{Spline}}$	Number of spline segments K	`{2,4,6,8}`
	Minimum slope $\varepsilon_{\text{slope}}$	`1e-4`
	Learning rate	`2e-3`
	Batch size (capped at the training-set size)	`{32,64,128}`
	Maximum epochs / patience	`300 / 80`
CoCoNet$_{\text{ICNN}}$	Convex-layer activation	`softplus`
	ICNN hidden dimension / depth	`64 / 2`
	Strong convexity $\varepsilon_{\text{ICNN}}$	`1e-3`
	Learning rate	`2e-3`
	Batch size	`32`
	Maximum epochs / patience	`400 / 80`
	M_{train} / M_{valid} / M_{test}	2^9 / 2^{12} / 2^{16}

For each dataset and each random seed, a frozen and shared point predictor $F(\mathbf{x})$, trained on $\mathcal{D}_{\text{pred}}$, is used to ensure a fair comparison between methods through identical residuals, so that the results reflect only the performance of the predicted uncertainty regions. Complete data partitioning, training configurations, and preprocessing details are shown in Table 4.

Computational Environment. Experiments are conducted using NVIDIA Quadro RTX 6000 GPU (24 GB).

CoCoNet Training and Hyperparameters. The two variants follow the same overall training pipeline, including the context encoder $\mathbf{h}(\mathbf{x})$, which takes the covariates together with the frozen point prediction $F(\mathbf{x})$, and use the same loss-weight search space. The score-specific architectures and optimization settings are selected separately based on each parameterization, as summarized in Table 5.

For *CoCoNet*$_{\text{ICNN}}$, the numerical volume estimator also requires a directional budget M, which determines the number of directions used to approximate the radial volume integral in (57). Different values of M are used across the training and evaluation stages: $M_{\text{train}} = 2^9$ for score optimization, $M_{\text{valid}} = 2^{12}$ for validation and model selection, and $M_{\text{test}} = 2^{16}$ for the final reported prediction-region sizes. A smaller budget is used during training to reduce the cost of repeated volume computations, while larger budgets are used for more accurate validation and final evaluation.

Table 6: Configurations and hyperparameter search spaces for the baseline methods. Search spaces are evaluated using only the validation portion of $\mathcal{D}_{\text{score}}$.

Method	**Hyperparameter**	**Value / Search Space**
Global Ellipsoid	Ridge	`1e-6`
Local Ellipsoid	Number of neighbors	$\max(2p+5, \lceil 0.05n \rceil)$
	Local / global covariance weight	`0.95 / 0.05`
	Ridge	`1e-6`
Empirical Copula	Normalizer hidden dimensions	`[64,32]`
	Scale offset	`0.1`
	Copula	`empirical diagonal copula`
MVCS	Center	$F(x)$
	Matrix hidden dimension	`256`
	Matrix depth	`{3,5}`
	Matrix learning rate	`{1e-4, 1e-3, 4e-3, 5e-3, 1e-2}`
	Norm-exponent learning rate	`{1e-3, 1e-2}`
	Norm exponent	`q` ≥ 1
	Batch size	`{100,200,1000}`
	Maximum epochs	`4000`
	Early stopping	`patience=500`
LW-SC	Hidden dimensions	`[64,32]`
	Learning rate	`1e-3`
	Batch size	`min(256,n)`
	Maximum epochs	`200`
	Early stopping	`patience=20`
PBS	(Width, depth, batch size)	`{(10,1,2048), (12,4,512), (16,4,512), (46,4,8192), (18,8,256)}`
	Optimizer / learning rate	`AdamW / 1e-2`
	Maximum epochs	`800`

Baseline Configurations. The baselines for multi-dimensional datasets already introduced in the main paper, although the primary porpuse of this study wasn't focusing on univariate prediction, however, For the univariate experiments, **Locally Weighted Split Conformal prediction (LW-SC)** (Lei et al., 2018) is additionally included. LW-SC fits a spread model $\hat{\sigma}(\mathbf{x})$ on $\mathcal{D}_{\text{score}}$ to predict the absolute residual magnitude and uses the scaled conformity score $|r|/\hat{\sigma}(\mathbf{x})$, producing symmetric prediction intervals whose width adapts to the input.

The extended multivariate comparison additionally includes **Pullback Conformal Sets (PBS)** (Kondratyev et al., 2025). The AC-NQR$_U$ variant is used because it operates on residuals relative to a fixed point predictor and is therefore compatible with the common $F(\mathbf{x})$ used across methods. PBS learns a conditional vector-quantile map of the residuals using neural optimal transport, with conformity score $\|Q^{-1}(\mathbf{r}, \mathbf{x})\|_2$.

All baseline implementations follow the configurations and model-selection procedures reported in their corresponding source papers. When multiple candidate settings are provided, the corresponding search spaces are evaluated using only the validation portion of $\mathcal{D}_{\text{score}}$. The exact configurations and search spaces used in the experiments are summarized in Table 6.

Hyperparameter Selection, Calibration, and Evaluation Protocol. For each dataset and random seed, all model and hyperparameter selection is performed using only $\mathcal{D}_{\text{score}}$. Candidate models are trained on its training portion, while the held-out validation portion is used for architecture, hyperparameter, and checkpoint selection. For *CoCoNet*, among candidate configurations and checkpoints with validation coverage above

$1-\alpha-\epsilon$, the one with the smallest validation region size is selected, with $\epsilon = 0.02$ in all reported experiments. For the baseline methods, hyperparameter and model selection is likewise restricted to the validation portion of $\mathcal{D}_{\text{score}}$.

After all model-selection decisions have been fixed, the selected conformity score is calibrated using the independent calibration set $\mathcal{D}_{\text{cal}}$. For each $(\mathbf{x}_i, \mathbf{y}_i) \in \mathcal{D}_{\text{cal}}$, define the calibration score $s_i = S(\mathbf{x}_i, \mathbf{y}_i - F(\mathbf{x}_i))$. The split-conformal threshold is then

$$\widehat{\tau}_\alpha = s_{(\lceil (n_{\text{cal}}+1)(1-\alpha) \rceil)}, \tag{68}$$

where $s_{(k)}$ denotes the kth order statistic of the calibration scores. The resulting prediction region for a new input $\mathbf{x}$ is

$$\widehat{\mathcal{C}}(\mathbf{x}) = \{\mathbf{y} : S(\mathbf{x}, \mathbf{y} - F(\mathbf{x})) \leq \widehat{\tau}_\alpha\}. \tag{69}$$

The test set $\mathcal{D}_{\text{test}}$ is used only after model selection and conformal calibration are complete. No hyperparameters, checkpoints, or calibration thresholds are adjusted using test-set performance. The same data splits are used across all compared methods for each random seed, and reported results are aggregated over the five seeds described in Table 4.

Evaluation Metrics. Prediction regions are evaluated primarily in terms of empirical coverage and size. For a test set of n_{test} observations, empirical coverage is computed as

$$\widehat{\text{Cov}} = \frac{1}{n_{\text{test}}} \sum_{i=1}^{n_{\text{test}}} \mathbb{1}\left\{\mathbf{y}_i \in \widehat{\mathcal{C}}(\mathbf{x}_i)\right\}. \tag{70}$$

Region size is summarized by the dimension-normalized size

$$\text{Size} = \left(\frac{1}{n_{\text{test}}} \sum_{i=1}^{n_{\text{test}}} V\left(\widehat{\mathcal{C}}(\mathbf{x}_i)\right) \right)^{1/p}, \tag{71}$$

where $V(\widehat{\mathcal{C}}(\mathbf{x}_i))$ denotes the p-dimensional volume of the prediction region. Taking the p-th root places volumes from different response dimensions on a comparable linear scale.

For *CoCoNet*$_{\text{ICNN}}$, the numerical stability of the final volume estimates is assessed using a budget-doubling diagnostic. Specifically, the dimension-normalized test-set size is evaluated using nested directional budgets $M = 2^{15}$ and $2M = 2^{16}$, with both estimates using prefixes of the same scrambled Sobol sequence. An estimate is considered stable when doubling the directional budget changes the dimension-normalized size by less than 0.1%. All reported *CoCoNet*$_{\text{ICNN}}$ results satisfy this stability criterion, and the values reported in the tables use the larger 2^{16}-direction estimate.

For fair evaluation, all reported prediction-region volumes in Table 2 are recomputed using the common numerical estimator in (57). During model training and selection, however, each method retains its native volume computation, including exact volume formulas when available; the common estimator is used only for the final cross-method comparison.

For the synthetic experiments, the known oracle region additionally permits direct evaluation of geometric recovery. Intersection over union is computed as

$$\text{IoU} = \frac{V\left(\widehat{\mathcal{C}} \cap \mathcal{C}^\star\right)}{V\left(\widehat{\mathcal{C}} \cup \mathcal{C}^\star\right)}. \tag{72}$$

where $\mathcal{C}^\star$ denotes the population-optimal region. Relative size is also reported through

$$\frac{V(\widehat{\mathcal{C}})}{V^\star}, \tag{73}$$

where $V^\star$ is the oracle volume. An IoU close to one indicates accurate geometric recovery, while a volume ratio close to one indicates that the learned region has a size close to the population optimum.

All reported metrics are computed on $\mathcal{D}_{\text{test}}$ after model selection and conformal calibration are complete. Results across real-data experiments are reported as mean $\pm$ standard deviation over the five random seeds.

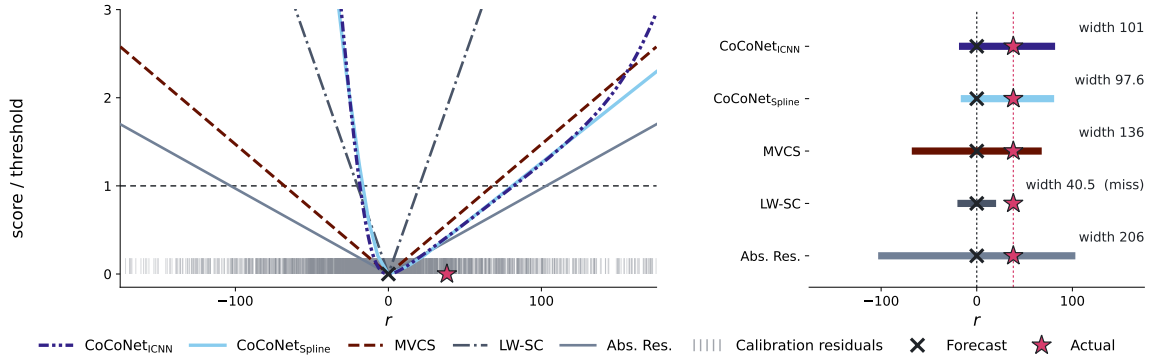

Figure 5: **Univariate conformity scores and prediction intervals on Beijing PM2.5** (one test observation). *Left:* conformity scores as functions of the residual $\mathbf{r} = \mathbf{y} - F(\mathbf{x})$, normalized by the calibrated threshold; gray ticks denote calibration residuals. *Right:* the corresponding 90% prediction intervals, with the forecast (cross) and actual observation (star). The *CoCoNet* variants learn asymmetric intervals, while MVCS and Absolute Residual remain symmetric; LW-SC gives the narrowest interval but misses the observation.

Table 7: **Univariate regression results at 90% target coverage.** Each method reports, in gray, the empirical test coverage in percent, followed by the prediction-region size (lower is better); subscripts are standard deviations over five seeds. Where a factor ($\times 10^k$) follows the dataset name, the sizes in that row are in units of 10^k. Best and second-best sizes are shown in **blue** and underline.

Dataset	Baselines						*CoCoNet* (this work)			
	Abs. Res.		LW-SC		MVCS		*CoCoNet*$_{\text{Spline}}$		*CoCoNet*$_{\text{ICNN}}$	
	Cov. %	Size	Cov. %	Size	Cov. %	Size	Cov. %	Size	Cov. %	Size
Beijing PM2.5 ($\times 10^2$)	$91.1_{\pm 0.8}$	$2.10_{\pm 0.05}$	$90.6_{\pm 1.1}$	$2.05_{\pm 0.12}$	$90.5_{\pm 1.1}$	$1.89_{\pm 0.07}$	$90.3_{\pm 0.6}$	$\mathbf{1.64_{\pm 0.02}}$	$90.7_{\pm 0.9}$	$\underline{1.68_{\pm 0.02}}$
Bike Sharing ($\times 10^2$)	$90.6_{\pm 0.6}$	$1.49_{\pm 0.05}$	$90.2_{\pm 1.2}$	$1.75_{\pm 0.23}$	$90.0_{\pm 1.4}$	$1.29_{\pm 0.02}$	$90.0_{\pm 0.9}$	$\mathbf{1.22_{\pm 0.03}}$	$90.5_{\pm 0.8}$	$\underline{1.28_{\pm 0.03}}$
CCPP ($\times 10^1$)	$90.1_{\pm 0.9}$	$1.28_{\pm 0.02}$	$90.1_{\pm 0.9}$	$1.28_{\pm 0.02}$	$90.4_{\pm 1.0}$	$1.25_{\pm 0.02}$	$90.2_{\pm 0.7}$	$\underline{1.24_{\pm 0.03}}$	$90.3_{\pm 0.7}$	$\mathbf{1.22_{\pm 0.01}}$
Concrete ($\times 10^1$)	$90.2_{\pm 1.5}$	$2.52_{\pm 0.21}$	$91.4_{\pm 2.8}$	$2.60_{\pm 0.46}$	$91.7_{\pm 1.9}$	$2.52_{\pm 0.06}$	$91.0_{\pm 1.7}$	$\mathbf{2.34_{\pm 0.15}}$	$92.4_{\pm 2.4}$	$\underline{2.49_{\pm 0.13}}$
Kin8nm	$91.1_{\pm 0.9}$	$0.30_{\pm 0.00}$	$90.4_{\pm 1.0}$	$0.30_{\pm 0.00}$	$90.7_{\pm 0.8}$	$0.31_{\pm 0.01}$	$90.2_{\pm 1.3}$	$\mathbf{0.29_{\pm 0.01}}$	$90.7_{\pm 1.0}$	$\underline{0.29_{\pm 0.01}}$
Protein ($\times 10^1$)	$90.8_{\pm 0.9}$	$1.56_{\pm 0.02}$	$90.8_{\pm 0.6}$	$1.46_{\pm 0.04}$	$90.8_{\pm 0.6}$	$1.35_{\pm 0.02}$	$90.8_{\pm 0.7}$	$\underline{1.26_{\pm 0.08}}$	$91.1_{\pm 0.6}$	$\mathbf{1.22_{\pm 0.01}}$
Mean rank ↓	$4.33_{\pm 0.82}$		$4.17_{\pm 0.75}$		$3.50_{\pm 0.84}$		$\mathbf{1.33_{\pm 0.52}}$		$\underline{1.67_{\pm 0.52}}$	

E Additional Experimental Results

Univariate Regression Results. While *CoCoNet* is primarily designed to construct efficient uncertainty regions for multi-dimensional outputs, it can also learn asymmetric conformity scores in the univariate setting. Table 7 reports the efficiency (i.e., interval width) and empirical coverage across six univariate datasets. For $p = 1$, the ℓ_1 and ℓ_∞ aggregations of a single coordinate score coincide, so *CoCoNet*$_{\text{Spline-}\ell_1}$ and *CoCoNet*$_{\text{Spline-}\ell_\infty}$ are identical and are reported as *CoCoNet*$_{\text{Spline}}$.

The results clearly show the efficiency of *CoCoNet* compared with the baselines. Abs. Res., which uses the basic symmetric residual score, has the lowest efficiency rank among the methods (4.33). LW-SC improves the rank to 4.17 by introducing an input-dependent scale for the residuals, showing the benefit of input adaptivity. MVCS further improves the efficiency rank to 3.50. Since norm-based regions reduce to intervals in one dimension, MVCS can also adapt the interval scale based on the input, providing stronger adaptivity than LW-SC. However, while both LW-SC and MVCS can use the input to adapt the scale of the score, neither can produce asymmetric scores.

In contrast, *CoCoNet* combines input-dependent adaptation with the ability to learn asymmetric score functions. This leads to substantially better efficiency ranks for both variants, with the two variants together producing the smallest intervals across all six datasets. In particular, *CoCoNet*$_{\text{Spline}}$ achieves an efficiency rank of 1.33,

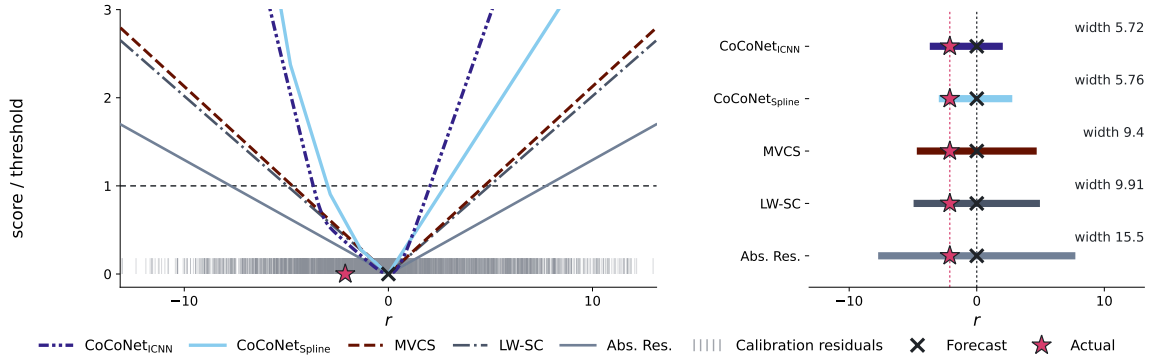

Figure 6: **Univariate conformity scores and prediction intervals on Protein** (one test observation). *Left:* conformity scores as functions of the residual $r = \mathbf{y} - F(\mathbf{x})$, normalized by the calibrated threshold; gray ticks denote calibration residuals. *Right:* the corresponding 90% prediction intervals, with the forecast (cross) and actual observation (star). *CoCoNet*$_{\text{ICNN}}$ and *CoCoNet*$_{\text{Spline}}$ produce the smallest intervals in this example.

while *CoCoNet*$_{\text{ICNN}}$ achieves a rank of 1.67, both substantially better than the baselines. These results show the benefit of learning asymmetric and flexible conformity scores even in the one-dimensional setting, and also indicate that *CoCoNet*$_{\text{Spline}}$ can be particularly effective for univariate regression.

Figures 5 and 6 further illustrate the numerical results using two example instances from two different datasets. The figures show how learning more flexible conformity scores can improve efficiency even when $p = 1$. In both examples, *CoCoNet*$_{\text{Spline}}$ and *CoCoNet*$_{\text{ICNN}}$ learn asymmetric score functions around the prediction, resulting in substantially smaller intervals than the other baselines. The empirical coverage of all methods remains close to the target coverage of 90%, indicating that the differences in interval size mainly come from the different conformity scores learned by each method.

Multivariate Results and Further Analysis. The main text reports the primary quantitative comparison across 30 multivariate regression tasks with response dimensions ranging from $p = 2$ to $p = 16$. This section complements those results by examining the learned prediction-region geometry, the consistency of efficiency across datasets, and the effect of expanding the comparison to more flexible MVCS variants and Pullback Conformal Sets (PBS). Together, these analyses provide a more detailed view of when the additional geometric flexibility of *CoCoNet* is beneficial and whether its efficiency persists beyond the convex structured baselines used in the primary comparison.

Qualitative Analysis of Learned Geometry. Figures 10 and 11 provide representative examples of the learned conformity-score geometry on two-dimensional multivariate tasks. Each figure shows *CoCoNet*$_{\text{ICNN}}$, *CoCoNet*$_{\text{Spline-}\ell_\infty}$, *CoCoNet*$_{\text{Spline-}\ell_1}$, MVCS, and Local Ellipsoid on the corresponding dataset, allowing the geometric behavior of *CoCoNet* to be compared with different structured alternatives. For each method, the left panel shows the calibration residuals and realized test residual, the middle panel visualizes the conformity score normalized by its calibrated threshold, and the right panel shows the resulting score contours and 90% prediction-region boundary. The two examples illustrate contrasting uncertainty structures: Online Retail II is strongly asymmetric, whereas Argoverse2 Motion is substantially closer to an elliptical geometry.

Figure 10 clearly reveals the advantage of learning the residual geometry through a flexible conformity score. The left panel shows a strongly asymmetric and non-ellipsoidal residual distribution, while the middle panel shows the learned score in the residual space. *CoCoNet*$_{\text{ICNN}}$ learns a curved, asymmetric, and convex score that closely reflects the residual distribution. Among the strong baselines, *CoCoNet*$_{\text{Spline-}\ell_\infty}$ is also capable of learning efficient asymmetric and rotated hyper-rectangular regions, and *CoCoNet*$_{\text{Spline-}\ell_1}$ learns asymmetric polygonal regions. In contrast, MVCS and Local Ellipsoid are not able to represent asymmetric norm-based or ellipsoidal regions, which leads to larger prediction regions. This is also consistent with the numerical results, where *CoCoNet*$_{\text{ICNN}}$ produces a prediction region with 7.2% smaller area compared to the next best method.

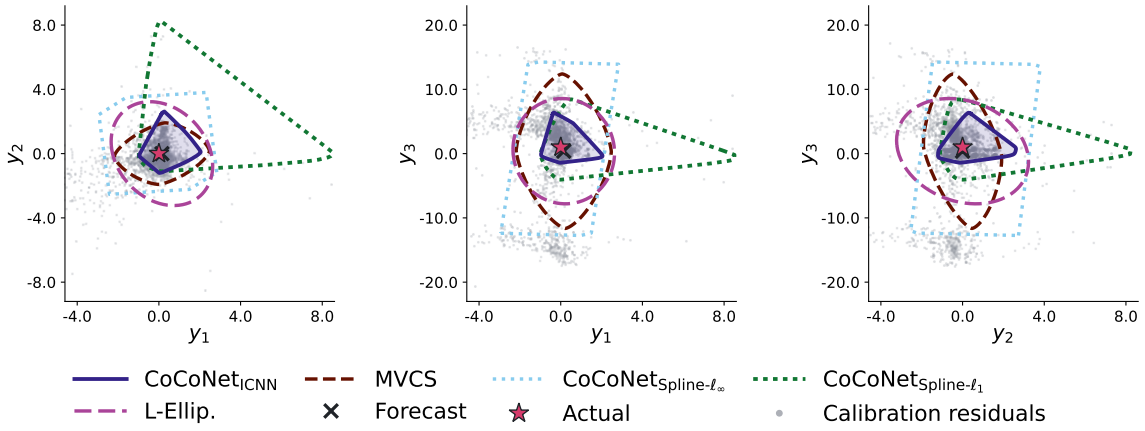

Figure 7: **Two-dimensional projections of three-dimensional prediction regions on Household Power** (one test observation). Each panel shows the exact projection of the 90% prediction region onto one pair of response coordinates, together with the point forecast (cross) and the observed response (star). The *CoCoNet*$_{\text{ICNN}}$ region adapts its shape to each pair of coordinates and is asymmetric around the forecast, whereas the MVCS region is centrally symmetric about the forecast and the *CoCoNet*$_{\text{Spline-}\ell_\infty}$ region is a rotated, asymmetric hyperrectangle whose projections are parallelograms; the *CoCoNet*$_{\text{Spline-}\ell_1}$ region is an asymmetric polytope.

Figure 11 shows a different setting, where the residual geometry is quasi-symmetric and closer to an ellipsoidal shape. This example further shows the stable performance of *CoCoNet*$_{\text{ICNN}}$ across both asymmetric and more structured geometries. In this setting, Local Ellipsoid and MVCS also produce efficient prediction regions, since their geometric assumptions are better aligned with the residual distribution. However, the additional flexibility of *CoCoNet*$_{\text{ICNN}}$ still provides an improvement, although with a smaller margin. In particular, *CoCoNet*$_{\text{ICNN}}$ produces a prediction region with about 1.8% smaller area than the next best method, showing its consistent efficiency across different uncertainty structures.

These examples further demonstrate the efficiency of *CoCoNet* across different uncertainty geometries. Rather than depending on a specific underlying shape, *CoCoNet* performs consistently across different settings. This is also supported by the synthetic results, where *CoCoNet*$_{\text{ICNN}}$ performs close to Local Ellipsoid and MVCS on the ellipsoidal example. In real datasets, however, the residual geometry is rarely a perfect ellipsoid. The ability of *CoCoNet*$_{\text{ICNN}}$ to capture the remaining asymmetry and local deviations allows it to improve over the more structured baselines even in these near-ellipsoidal settings.

Higher-Dimensional Projection Analysis. Since prediction regions cannot be directly visualized in higher dimensions ($p > 2$), Figures 7 and 8 show two-dimensional projections of the learned regions for Household Power ($p = 3$) and QM8 ($p = 4$). The residual distributions again show asymmetric structures across different coordinate pairs, and *CoCoNet*$_{\text{ICNN}}$ is able to adapt its geometry accordingly, learning different shapes across the projections. In comparison, MVCS remains centrally symmetric, while *CoCoNet*$_{\text{Spline-}\ell_\infty}$ can learn asymmetric rotated hyper-rectangular regions and *CoCoNet*$_{\text{Spline-}\ell_1}$ asymmetric polytopes. This behavior is also reflected in the numerical results, where *CoCoNet*$_{\text{ICNN}}$ produces an approximately 18.5% smaller prediction-region volume on Household Power and approximately 42.0% smaller volume on QM8 compared with the next best method.

Consistency Across Multivariate Tasks. Figure 9 summarizes the efficiency ranks of all methods across the 30 multivariate regression tasks. For each dataset, the methods are ranked based on their dimension-normalized prediction-region size, where a lower rank represents better efficiency. This figure provides a better view of the consistency of each method across the different datasets, complementing the individual results reported in Table 2.

The results show a clear and consistent advantage for *CoCoNet*$_{\text{ICNN}}$, with a mean rank of 1.45. As already mentioned, *CoCoNet*$_{\text{ICNN}}$ is the most efficient method on 22 out of the 30 datasets, while it remains among the

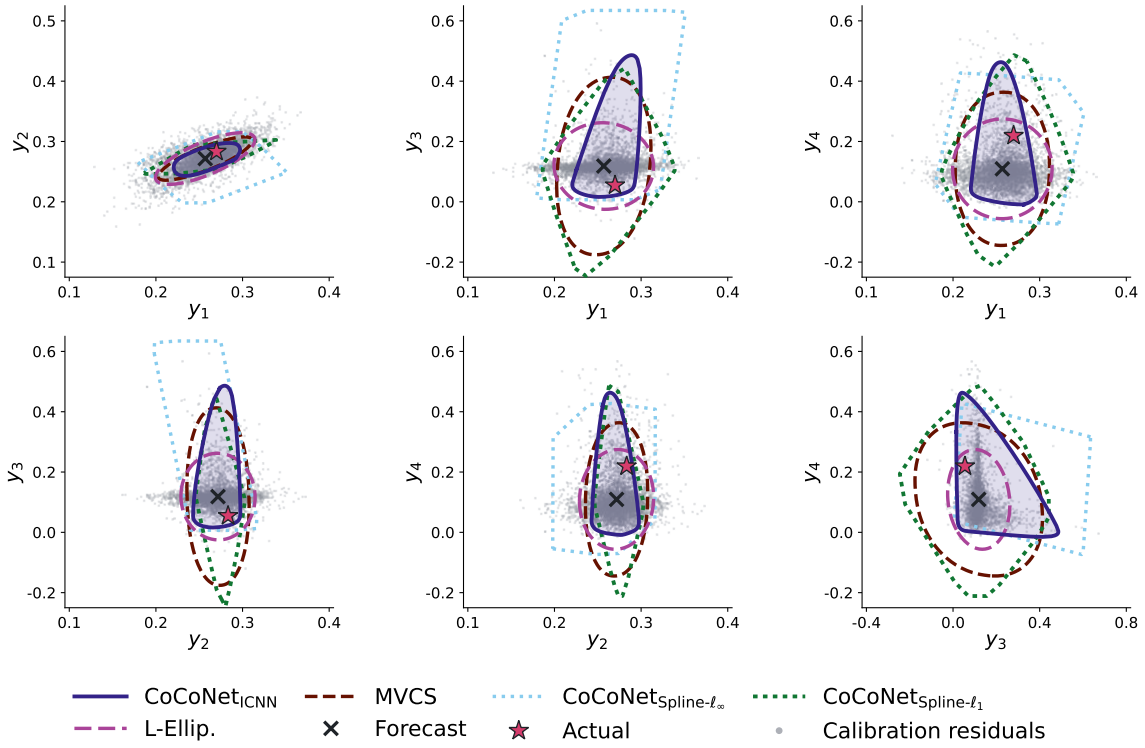

Figure 8: **Two-dimensional projections of four-dimensional prediction regions on QM8** (one test observation). Each panel shows the exact projection of the 90% prediction region onto one pair of response coordinates, together with the point forecast (cross), the observed response (star), and the calibration residuals centered at the point forecast (gray points). The *CoCoNet*$_{\text{ICNN}}$ region adapts its shape to each pair of coordinates and is asymmetric around the forecast, whereas the MVCS region is centrally symmetric about the forecast and the *CoCoNet*$_{\text{Spline-}\ell_\infty}$ region is a rotated, asymmetric hyperrectangle whose projections are polygons with parallel opposite edges; the *CoCoNet*$_{\text{Spline-}\ell_1}$ region is an asymmetric polytope.

three best-performing methods on 29. As discussed in the previous section, this shows the stable performance of *CoCoNet*$_{\text{ICNN}}$ across different application areas and response dimensions. *CoCoNet*$_{\text{Spline-}\ell_1}$ provides the second-best mean rank of 2.75 and ranks first on four other tasks, followed by MVCS with 3.58, while *CoCoNet*$_{\text{Spline-}\ell_\infty}$ achieves a mean rank of 4.28 and ranks first on two tasks. Among the remaining two tasks, MVCS and Local Ellipsoid each rank first once.

The other methods show wider rank distributions, indicating less consistent performance across the datasets. Local and Global Ellipsoid achieve mean ranks of 4.47 and 5.15, respectively, while Absolute Residual and Empirical Copula obtain mean ranks of 7.18 and 7.13. Overall, these results further show that the efficiency of *CoCoNet*$_{\text{ICNN}}$ is consistent across the benchmark and is not limited to a specific dataset, response dimension, or uncertainty geometry.

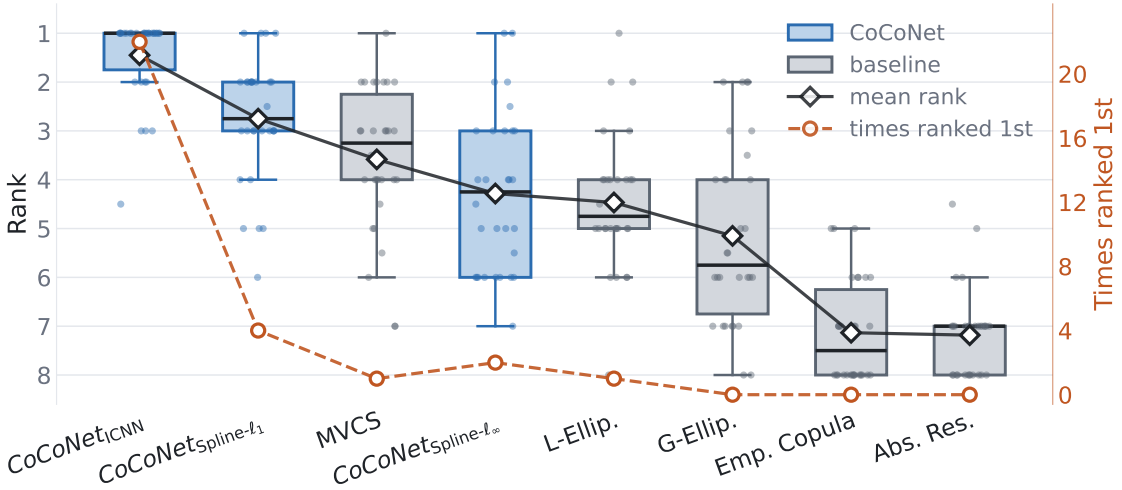

Figure 9: **Distribution of prediction-region efficiency ranks across 30 multivariate regression tasks.** Lower rank indicates greater efficiency. Boxplots show the distribution of per-dataset ranks, diamonds denote mean rank, and the dashed curve reports the number of tasks on which each method ranks first.

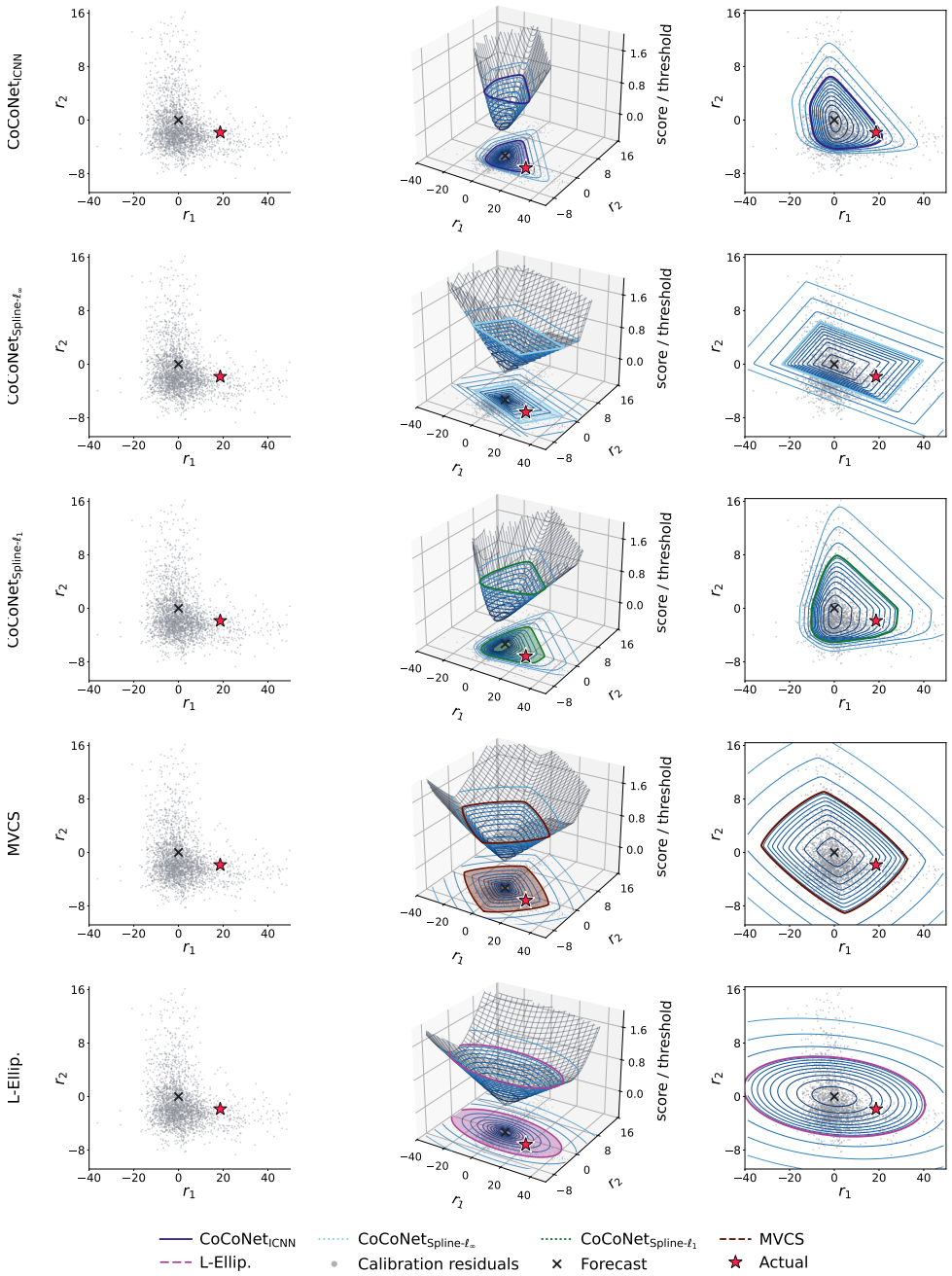

Figure 10: **Learned conformity scores and prediction regions on Online Retail II.** Each row shows one method at the same test input. *Left:* calibration residuals $\mathbf{r} = \mathbf{y} - F(\mathbf{x})$ with the point forecast (cross, at the origin) and the actual (star). *Middle:* the learned conformity score over the residual space, normalized by its calibrated threshold, so that the 90% prediction region is the sublevel set at height one; the region is traced on the surface and projected, with the score level curves, onto the floor. *Right:* level curves of the score together with the boundary of the prediction region. The residuals are strongly skewed; the *CoCoNet*$_{\text{ICNN}}$ score bends its level sets toward the mass of the residuals and yields an asymmetric region, whereas the *CoCoNet*$_{\text{Spline-}\ell_\infty}$ region is a rotated rectangle, the *CoCoNet*$_{\text{Spline-}\ell_1}$ region an asymmetric polygon, and the MVCS and Local Ellipsoid regions remains centrally symmetric about the forecast, so both must cover low-density areas to reach the target coverage.

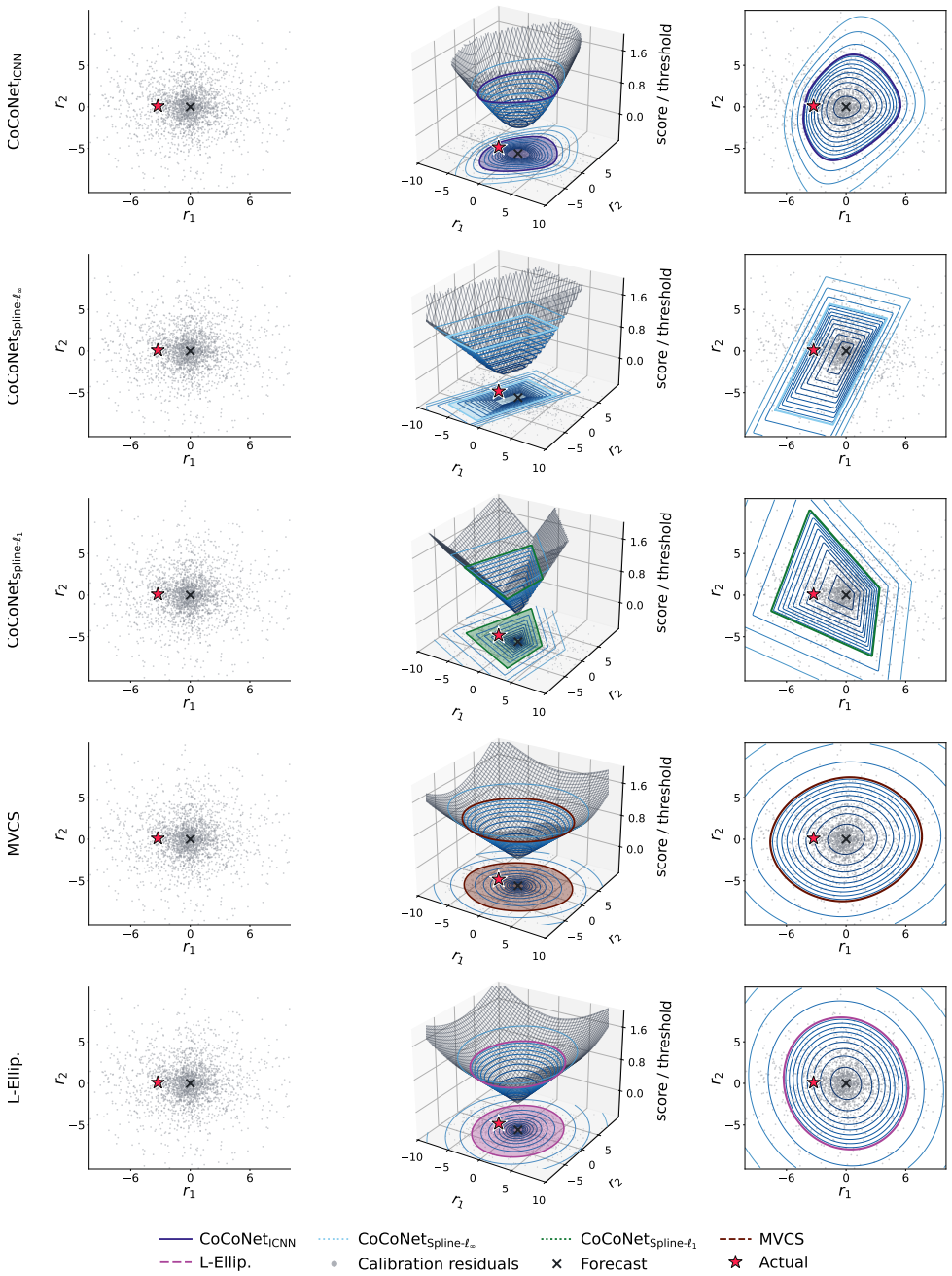

Figure 11: **Learned conformity scores and prediction regions on Argoverse2 Motion.** Each row shows one method at the same test input. *Left:* calibration residuals $\mathbf{r} = \mathbf{y} - F(\mathbf{x})$ with the point forecast (cross, at the origin) and the realized residual (star). *Middle:* the learned conformity score over the residual space, normalized by its calibrated threshold, so that the 90% prediction region is the sublevel set at height one; the region is traced on the surface and projected, with the score level curves, onto the floor. *Right:* level curves of the score together with the boundary of the prediction region. Here the residuals are close to elliptical and the three regions are similar in extent; the *CoCoNet*$_{\text{ICNN}}$ region departs from an ellipse only where the residual density does, while the *CoCoNet*$_{\text{Spline-}\ell_\infty}$ and Local Ellipsoid regions keep the rectangular and elliptical shapes imposed by their score families, and the *CoCoNet*$_{\text{Spline-}\ell_1}$ region stays polygonal.

Table 8: **Extended multivariate comparison with MVCS variants and PBS at 90% target coverage.** Each method reports empirical test coverage in gray followed by the dimension-normalized prediction-region size (lower is better); subscripts denote standard deviations over five seeds. MVCS variants with $q \geq 1$ preserve convexity, whereas the free-q variants do not impose this restriction. Where a factor ($\times 10^k$) follows the dataset name, reported sizes are scaled by 10^k. The best and second-best sizes are shown in **blue** and underlined.

		Baselines										*CoCoNet*					
Dataset	p	$\text{MVCS}_{q\geq 1}$		$\text{MVCS}_{\text{free } q}$		$\text{MVCS}_{\text{multi, } q\geq 1}$		$\text{MVCS}_{\text{multi, free } q}$		PBS		Spline-ℓ_∞		Spline-ℓ_1		ICNN	
		Cov. %	Size	Cov. %	Size	Cov. %	Size	Cov. %	Size	Cov. %	Size	Cov. %	Size	Cov. %	Size	Cov. %	Size
Argoverse2-2D ($\times 10^1$)	2	$90.2_{\pm 0.7}$	$3.68_{\pm 0.08}$	$90.1_{\pm 0.6}$	$3.68_{\pm 0.05}$	$90.0_{\pm 0.4}$	$3.89_{\pm 0.03}$	$89.9_{\pm 0.5}$	$3.89_{\pm 0.03}$	$90.2_{\pm 0.6}$	$3.99_{\pm 0.11}$	$90.1_{\pm 0.6}$	$3.58_{\pm 0.03}$	$90.2_{\pm 0.7}$	$\mathbf{3.50_{\pm 0.10}}$	$89.8_{\pm 0.6}$	$3.51_{\pm 0.06}$
Argoverse2 Motion ($\times 10^1$)	2	$90.0_{\pm 0.7}$	$1.14_{\pm 0.03}$	$90.1_{\pm 0.8}$	$1.14_{\pm 0.03}$	$90.3_{\pm 0.5}$	$1.20_{\pm 0.01}$	$90.3_{\pm 0.5}$	$1.20_{\pm 0.01}$	$90.1_{\pm 0.8}$	$1.22_{\pm 0.02}$	$90.1_{\pm 0.6}$	$1.11_{\pm 0.02}$	$90.3_{\pm 0.9}$	$1.11_{\pm 0.03}$	$89.9_{\pm 0.6}$	$\mathbf{1.10_{\pm 0.01}}$
Bias-Corr. Temp.	2	$89.8_{\pm 0.8}$	$4.08_{\pm 0.08}$	$89.9_{\pm 1.0}$	$4.06_{\pm 0.05}$	$89.8_{\pm 0.9}$	$4.09_{\pm 0.04}$	$89.8_{\pm 0.9}$	$4.09_{\pm 0.04}$	$90.2_{\pm 0.9}$	$4.21_{\pm 0.07}$	$89.7_{\pm 0.8}$	$3.86_{\pm 0.10}$	$89.9_{\pm 0.9}$	$\mathbf{3.82_{\pm 0.13}}$	$90.4_{\pm 0.5}$	$3.88_{\pm 0.13}$
ENB	2	$90.0_{\pm 2.5}$	$9.56_{\pm 2.32}$	$90.6_{\pm 3.1}$	$9.28_{\pm 1.13}$	$89.1_{\pm 4.5}$	$10.80_{\pm 1.05}$	$89.0_{\pm 4.4}$	$11.15_{\pm 1.48}$	$89.7_{\pm 4.2}$	$8.00_{\pm 0.44}$	$90.3_{\pm 3.0}$	$8.14_{\pm 1.24}$	$90.5_{\pm 1.8}$	$8.19_{\pm 0.39}$	$89.4_{\pm 3.5}$	$\mathbf{7.55_{\pm 0.13}}$
Geographical Music ($\times 10^2$)	2	$89.1_{\pm 2.6}$	$1.26_{\pm 0.15}$	$89.6_{\pm 2.4}$	$1.24_{\pm 0.12}$	$90.5_{\pm 0.9}$	$1.08_{\pm 0.09}$	$90.6_{\pm 0.9}$	$1.09_{\pm 0.08}$	$90.2_{\pm 1.1}$	$1.33_{\pm 0.06}$	$90.7_{\pm 1.6}$	$\mathbf{1.06_{\pm 0.02}}$	$91.5_{\pm 2.1}$	$1.08_{\pm 0.04}$	$91.2_{\pm 2.0}$	$1.14_{\pm 0.13}$
CASP ($\times 10^1$)	2	$89.8_{\pm 0.6}$	$2.48_{\pm 0.08}$	$89.8_{\pm 0.8}$	$2.47_{\pm 0.05}$	$90.1_{\pm 0.6}$	$2.85_{\pm 0.04}$	$90.2_{\pm 0.6}$	$2.85_{\pm 0.04}$	$90.2_{\pm 0.9}$	$2.90_{\pm 0.05}$	$89.7_{\pm 0.9}$	$2.42_{\pm 0.06}$	$89.7_{\pm 0.4}$	$2.39_{\pm 0.08}$	$89.8_{\pm 0.8}$	$\mathbf{2.34_{\pm 0.09}}$
Online Retail II ($\times 10^1$)	2	$89.9_{\pm 0.4}$	$2.25_{\pm 0.12}$	$89.9_{\pm 0.4}$	$2.28_{\pm 0.12}$	$89.8_{\pm 0.5}$	$2.18_{\pm 0.35}$	$89.5_{\pm 0.5}$	$2.17_{\pm 0.37}$	$90.1_{\pm 0.4}$	$3.68_{\pm 0.49}$	$89.7_{\pm 0.4}$	$2.05_{\pm 0.06}$	$89.8_{\pm 0.6}$	$1.92_{\pm 0.11}$	$90.0_{\pm 0.5}$	$\mathbf{1.85_{\pm 0.12}}$
QM9-2D	2	$89.8_{\pm 0.5}$	$5.30_{\pm 0.06}$	$89.5_{\pm 0.6}$	$5.26_{\pm 0.07}$	$89.3_{\pm 0.6}$	$5.73_{\pm 0.05}$	$89.3_{\pm 0.6}$	$5.72_{\pm 0.06}$	$90.0_{\pm 0.5}$	$5.65_{\pm 0.18}$	$89.6_{\pm 0.4}$	$\mathbf{5.06_{\pm 0.07}}$	$89.4_{\pm 0.4}$	$5.10_{\pm 0.10}$	$89.5_{\pm 0.8}$	$5.08_{\pm 0.11}$
Household Power	3	$89.9_{\pm 0.5}$	$5.46_{\pm 0.37}$	$89.9_{\pm 0.5}$	$5.43_{\pm 0.35}$	$90.1_{\pm 0.6}$	$12.08_{\pm 1.48}$	$90.1_{\pm 0.6}$	$12.02_{\pm 1.42}$	$89.9_{\pm 0.2}$	$15.64_{\pm 1.07}$	$90.1_{\pm 0.4}$	$7.15_{\pm 0.71}$	$90.2_{\pm 0.4}$	$6.96_{\pm 0.86}$	$89.9_{\pm 0.4}$	$\mathbf{5.10_{\pm 0.66}}$
SCPF	3	$90.9_{\pm 1.1}$	$5.82_{\pm 0.32}$	$90.7_{\pm 1.6}$	$5.71_{\pm 0.25}$	$89.3_{\pm 1.9}$	$7.33_{\pm 0.60}$	$89.3_{\pm 1.9}$	$7.33_{\pm 0.60}$	$90.9_{\pm 2.0}$	$9.50_{\pm 2.60}$	$89.6_{\pm 1.5}$	$6.07_{\pm 0.99}$	$88.2_{\pm 3.1}$	$5.19_{\pm 0.80}$	$89.0_{\pm 3.2}$	$\mathbf{5.16_{\pm 0.68}}$
Argoverse2-4D ($\times 10^1$)	4	$90.1_{\pm 0.5}$	$1.61_{\pm 0.02}$	$90.3_{\pm 0.4}$	$1.61_{\pm 0.02}$	$90.5_{\pm 0.6}$	$2.18_{\pm 0.10}$	$90.5_{\pm 0.6}$	$2.18_{\pm 0.10}$	$89.7_{\pm 0.7}$	$1.77_{\pm 0.11}$	$89.9_{\pm 0.3}$	$1.64_{\pm 0.02}$	$90.2_{\pm 0.4}$	$1.60_{\pm 0.04}$	$89.9_{\pm 0.6}$	$\mathbf{1.57_{\pm 0.02}}$
QM8 ($\times 10^{-2}$)	4	$89.4_{\pm 0.6}$	$11.89_{\pm 0.09}$	$89.6_{\pm 0.6}$	$11.94_{\pm 0.13}$	$89.6_{\pm 1.1}$	$15.41_{\pm 0.47}$	$89.6_{\pm 1.1}$	$15.41_{\pm 0.47}$	$90.0_{\pm 0.4}$	$13.74_{\pm 0.14}$	$89.7_{\pm 0.8}$	$11.37_{\pm 0.23}$	$89.6_{\pm 0.3}$	$11.07_{\pm 0.09}$	$89.3_{\pm 0.8}$	$\mathbf{9.66_{\pm 0.21}}$
QM9-4D	4	$89.6_{\pm 0.5}$	$3.48_{\pm 0.04}$	$89.6_{\pm 0.7}$	$3.48_{\pm 0.05}$	$89.5_{\pm 0.5}$	$4.15_{\pm 0.14}$	$89.5_{\pm 0.5}$	$4.15_{\pm 0.14}$	$89.7_{\pm 0.4}$	$4.01_{\pm 0.33}$	$89.8_{\pm 0.2}$	$3.55_{\pm 0.04}$	$89.6_{\pm 0.8}$	$3.47_{\pm 0.03}$	$89.9_{\pm 0.6}$	$\mathbf{3.34_{\pm 0.02}}$
SGEMM ($\times 10^1$)	4	$89.9_{\pm 0.5}$	$3.03_{\pm 0.17}$	$90.1_{\pm 0.6}$	$\mathbf{3.02_{\pm 0.13}}$	$90.0_{\pm 0.9}$	$9.51_{\pm 0.86}$	$90.0_{\pm 0.9}$	$9.46_{\pm 0.83}$	$89.8_{\pm 0.5}$	$16.08_{\pm 1.10}$	$90.1_{\pm 0.4}$	$3.22_{\pm 0.20}$	$90.1_{\pm 0.6}$	$3.08_{\pm 0.16}$	$90.1_{\pm 0.4}$	$3.02_{\pm 0.18}$
Beijing AQ-6D ($\times 10^2$)	6	$89.7_{\pm 0.5}$	$1.90_{\pm 0.03}$	$89.4_{\pm 0.6}$	$1.89_{\pm 0.04}$	$89.6_{\pm 0.4}$	$2.53_{\pm 0.06}$	$89.6_{\pm 0.4}$	$2.53_{\pm 0.06}$	$90.1_{\pm 0.6}$	$2.29_{\pm 0.24}$	$89.9_{\pm 0.5}$	$1.97_{\pm 0.02}$	$89.9_{\pm 0.3}$	$1.82_{\pm 0.01}$	$89.9_{\pm 0.6}$	$\mathbf{1.68_{\pm 0.02}}$
Communities Crime ($\times 10^2$)	6	$91.7_{\pm 1.1}$	$6.30_{\pm 0.35}$	$91.4_{\pm 1.3}$	$6.16_{\pm 0.07}$	$91.3_{\pm 1.6}$	$6.76_{\pm 0.33}$	$91.3_{\pm 1.6}$	$6.76_{\pm 0.33}$	$90.8_{\pm 2.1}$	$8.28_{\pm 0.42}$	$91.6_{\pm 1.7}$	$6.94_{\pm 0.54}$	$90.6_{\pm 2.2}$	$6.07_{\pm 0.10}$	$90.6_{\pm 1.7}$	$\mathbf{5.94_{\pm 0.17}}$
ATP1D ($\times 10^2$)	6	$89.4_{\pm 3.1}$	$4.61_{\pm 1.07}$	$89.1_{\pm 3.6}$	$4.61_{\pm 1.13}$	$90.3_{\pm 3.8}$	$5.55_{\pm 1.03}$	$90.3_{\pm 3.8}$	$5.55_{\pm 1.03}$	$90.9_{\pm 3.0}$	$4.18_{\pm 0.45}$	$89.4_{\pm 5.5}$	$4.20_{\pm 0.47}$	$88.5_{\pm 4.9}$	$4.50_{\pm 0.91}$	$88.8_{\pm 3.8}$	$\mathbf{3.60_{\pm 0.43}}$
QM9-6D	6	$90.0_{\pm 0.7}$	$5.07_{\pm 0.06}$	$89.8_{\pm 0.9}$	$5.06_{\pm 0.04}$	$89.5_{\pm 0.7}$	$7.25_{\pm 0.32}$	$89.5_{\pm 0.7}$	$7.25_{\pm 0.32}$	$90.1_{\pm 0.7}$	$5.75_{\pm 0.19}$	$90.0_{\pm 0.5}$	$5.35_{\pm 0.07}$	$89.8_{\pm 1.0}$	$4.90_{\pm 0.08}$	$90.1_{\pm 0.6}$	$\mathbf{4.85_{\pm 0.11}}$
SARCOS	7	$89.7_{\pm 0.5}$	$3.61_{\pm 0.05}$	$89.9_{\pm 0.6}$	$3.63_{\pm 0.02}$	$89.9_{\pm 0.8}$	$4.66_{\pm 0.06}$	$89.9_{\pm 0.8}$	$4.66_{\pm 0.06}$	$90.0_{\pm 0.9}$	$4.44_{\pm 0.39}$	$89.9_{\pm 0.8}$	$3.87_{\pm 0.07}$	$89.9_{\pm 0.4}$	$3.48_{\pm 0.09}$	$89.8_{\pm 0.4}$	$\mathbf{3.34_{\pm 0.01}}$
Argoverse2-8D	8	$90.1_{\pm 0.4}$	$6.84_{\pm 0.11}$	$89.9_{\pm 0.4}$	$6.84_{\pm 0.15}$	$90.2_{\pm 0.4}$	$16.69_{\pm 1.00}$	$90.2_{\pm 0.4}$	$16.69_{\pm 1.00}$	$90.0_{\pm 0.4}$	$9.20_{\pm 0.87}$	$90.0_{\pm 0.8}$	$7.83_{\pm 0.24}$	$90.0_{\pm 0.7}$	$6.96_{\pm 0.12}$	$90.1_{\pm 0.6}$	$\mathbf{6.76_{\pm 0.10}}$
QM9-8D	8	$90.2_{\pm 0.8}$	$4.62_{\pm 0.09}$	$90.2_{\pm 0.7}$	$4.61_{\pm 0.06}$	$89.8_{\pm 0.5}$	$7.80_{\pm 0.32}$	$89.8_{\pm 0.5}$	$7.80_{\pm 0.32}$	$89.6_{\pm 0.8}$	$5.24_{\pm 0.23}$	$89.9_{\pm 0.5}$	$5.06_{\pm 0.09}$	$89.9_{\pm 0.7}$	$\mathbf{4.38_{\pm 0.09}}$	$90.1_{\pm 0.9}$	$4.41_{\pm 0.09}$
Beijing AQ-12D ($\times 10^2$)	12	$89.7_{\pm 0.3}$	$1.34_{\pm 0.03}$	$89.7_{\pm 0.3}$	$1.33_{\pm 0.03}$	$89.8_{\pm 0.8}$	$2.56_{\pm 0.11}$	$89.8_{\pm 0.8}$	$2.56_{\pm 0.11}$	$90.0_{\pm 0.7}$	$2.92_{\pm 0.52}$	$89.5_{\pm 0.2}$	$1.55_{\pm 0.02}$	$89.7_{\pm 0.6}$	$\mathbf{1.31_{\pm 0.01}}$	$89.4_{\pm 0.3}$	$1.33_{\pm 0.05}$
Knee Contact Force ($\times 10^1$)	12	$90.1_{\pm 1.8}$	$6.81_{\pm 0.06}$	$90.0_{\pm 1.9}$	$6.68_{\pm 0.22}$	$89.9_{\pm 1.9}$	$16.55_{\pm 0.58}$	$89.9_{\pm 1.9}$	$16.55_{\pm 0.58}$	$90.5_{\pm 0.7}$	$9.06_{\pm 0.36}$	$90.4_{\pm 1.2}$	$8.27_{\pm 0.22}$	$89.8_{\pm 1.6}$	$7.16_{\pm 0.11}$	$90.3_{\pm 2.3}$	$\mathbf{6.65_{\pm 0.33}}$
WQ	14	$89.4_{\pm 2.5}$	$\mathbf{5.95_{\pm 0.27}}$	$90.1_{\pm 2.2}$	$6.39_{\pm 0.35}$	$89.2_{\pm 3.4}$	$6.36_{\pm 0.24}$	$89.2_{\pm 3.4}$	$6.36_{\pm 0.24}$	$87.7_{\pm 2.4}$	$7.21_{\pm 0.71}$	$89.0_{\pm 1.8}$	$6.93_{\pm 0.19}$	$89.3_{\pm 1.4}$	$6.28_{\pm 0.53}$	$87.9_{\pm 2.1}$	$6.11_{\pm 0.26}$
QM7b	14	$90.4_{\pm 0.7}$	$1.88_{\pm 0.03}$	$90.0_{\pm 0.9}$	$1.89_{\pm 0.04}$	$90.5_{\pm 0.5}$	$4.24_{\pm 0.13}$	$90.5_{\pm 0.5}$	$4.24_{\pm 0.13}$	$90.4_{\pm 0.6}$	$2.54_{\pm 0.07}$	$90.5_{\pm 0.7}$	$2.32_{\pm 0.02}$	$89.9_{\pm 1.2}$	$1.97_{\pm 0.03}$	$90.1_{\pm 0.6}$	$\mathbf{1.78_{\pm 0.03}}$
OES10 ($\times 10^3$)	16	$90.2_{\pm 5.4}$	$2.77_{\pm 0.68}$	$90.5_{\pm 5.0}$	$2.77_{\pm 0.67}$	$90.8_{\pm 5.6}$	$2.81_{\pm 0.66}$	$90.8_{\pm 5.6}$	$2.81_{\pm 0.66}$	$90.0_{\pm 5.8}$	$\mathbf{1.81_{\pm 0.30}}$	$90.8_{\pm 4.7}$	$3.86_{\pm 1.13}$	$90.5_{\pm 4.4}$	$2.95_{\pm 0.64}$	$91.0_{\pm 4.8}$	$2.39_{\pm 0.51}$
OES97 ($\times 10^3$)	16	$92.4_{\pm 4.7}$	$4.85_{\pm 1.16}$	$91.8_{\pm 4.3}$	$4.16_{\pm 0.98}$	$90.6_{\pm 2.9}$	$3.26_{\pm 0.26}$	$90.6_{\pm 2.9}$	$3.26_{\pm 0.26}$	$91.8_{\pm 3.5}$	$39.09_{\pm 16.09}$	$90.0_{\pm 3.9}$	$4.63_{\pm 0.93}$	$91.5_{\pm 3.9}$	$3.16_{\pm 0.42}$	$92.7_{\pm 2.2}$	$\mathbf{2.87_{\pm 0.29}}$
BDG2-Fox ($\times 10^1$)	16	$89.5_{\pm 0.9}$	$7.72_{\pm 0.11}$	$89.8_{\pm 0.7}$	$7.73_{\pm 0.15}$	$89.7_{\pm 0.5}$	$9.90_{\pm 0.16}$	$89.7_{\pm 0.5}$	$9.90_{\pm 0.16}$	$89.8_{\pm 0.3}$	$9.27_{\pm 0.18}$	$89.9_{\pm 0.7}$	$9.73_{\pm 0.31}$	$89.7_{\pm 0.8}$	$10.97_{\pm 1.65}$	$89.7_{\pm 1.2}$	$\mathbf{7.54_{\pm 0.16}}$
BDG2-Eagle ($\times 10^1$)	16	$89.8_{\pm 0.5}$	$7.15_{\pm 0.06}$	$90.0_{\pm 0.3}$	$7.18_{\pm 0.05}$	$90.0_{\pm 0.6}$	$8.36_{\pm 0.11}$	$90.0_{\pm 0.6}$	$8.36_{\pm 0.11}$	$89.8_{\pm 0.6}$	$7.97_{\pm 0.24}$	$89.4_{\pm 0.8}$	$8.99_{\pm 0.16}$	$89.7_{\pm 0.4}$	$7.35_{\pm 0.08}$	$89.8_{\pm 1.1}$	$\mathbf{7.13_{\pm 0.12}}$
BDG2-Lamb ($\times 10^1$)	16	$89.6_{\pm 0.7}$	$7.31_{\pm 0.12}$	$89.8_{\pm 0.7}$	$7.34_{\pm 0.12}$	$89.8_{\pm 0.2}$	$12.11_{\pm 0.15}$	$89.8_{\pm 0.2}$	$12.11_{\pm 0.15}$	$89.8_{\pm 0.8}$	$9.23_{\pm 0.15}$	$90.2_{\pm 0.6}$	$8.95_{\pm 0.20}$	$89.6_{\pm 0.4}$	$7.51_{\pm 0.15}$	$89.6_{\pm 0.7}$	$\mathbf{7.19_{\pm 0.25}}$
Mean rank ↓		3.92 ± 1.46		3.75 ± 1.28		6.58 ± 1.27		6.53 ± 1.19		6.33 ± 1.92		4.52 ± 1.74		2.93 ± 1.61		$\mathbf{1.43 \pm 0.85}$	

Table 9: **Structural properties of the MVCS variants and *CoCoNet*$_{\text{ICNN}}$.** Single-norm MVCS remains centrally symmetric for both constrained and free q, whereas the multi-norm formulation can represent asymmetric regions. The evaluated multi-norm variants use the non-input-dependent formulation of MVCS Braun et al. (2026). The $q \geq 1$ variants preserve convexity, while *CoCoNet*$_{\text{ICNN}}$ is input-dependent, asymmetric, and convex by construction.

Method	Input-dependent	Asymmetric	Convex guarantee
$\text{MVCS}_{q\geq 1}$	Yes	No	Yes
$\text{MVCS}_{\text{free } q}$	Yes	No	No
$\text{MVCS}_{\text{multi, } q\geq 1}$	No	Yes	Yes
$\text{MVCS}_{\text{multi, free } q}$	No	Yes	No
CoCoNet$_{\text{ICNN}}$	Yes	Yes	Yes

Extended Comparison with MVCS Variants. Table 2 reports the MVCS results using the single-norm variant with the convexity restriction $q \geq 1$. This setting was selected to provide a fair comparison across the convex baselines. Here, the study further compares *CoCoNet* with other variants of MVCS to examine whether its efficiency advantage remains when the convexity restriction is removed. Table 9 summarizes the properties

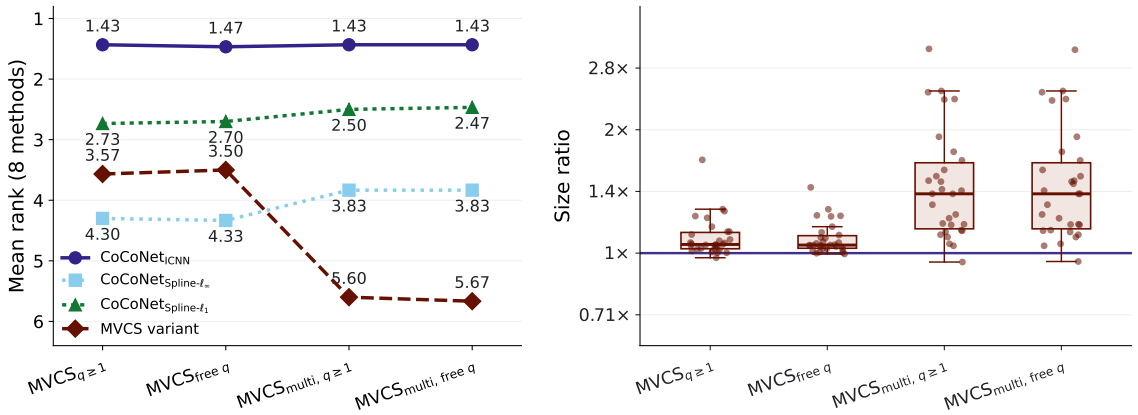

Figure 12: **MVCS variants versus *CoCoNet*.** *Left:* mean rank of *CoCoNet*$_{\text{ICNN}}$, *CoCoNet*$_{\text{Spline-}\ell_\infty}$, *CoCoNet*$_{\text{Spline-}\ell_1}$, and the corresponding MVCS variant across the 30 multivariate regression tasks, with one MVCS variant included at a time in the primary eight-method comparison (lower is better). *Right:* dimension-normalized prediction-region size of each MVCS variant relative to *CoCoNet*$_{\text{ICNN}}$ on each task (log scale); the horizontal line marks equal size, boxes show quartiles, whiskers extend to 1.5 times the interquartile range, and points denote individual tasks. The label $q \geq 1$ denotes variants whose norm exponents are constrained to preserve convexity.

of the evaluated MVCS variants, covering the four combinations of single- and multi-norm formulations with constrained and unrestricted norm exponents.

Removing the restriction allows nonconvex quasi-norm regions and increases the flexibility of MVCS. However, the single-norm variants remain centrally symmetric around their center, even when q is unrestricted. Therefore, relaxing q can change the curvature and sparsity of the boundary, but it still cannot capture asymmetric uncertainty geometries. The multi-norm extension of Braun et al. (2026) further increases the flexibility by using different norm parameters across different regions of the space, which removes this symmetry restriction.

This distinction helps explain the relatively modest effect of removing the $q \geq 1$ constraint in the present experiments. The unrestricted single-norm variant improves the aggregate mean rank from 3.27 to 3.00, but retains the same central-symmetry restriction. In contrast, *CoCoNet*$_{\text{ICNN}}$ can represent asymmetric convex geometries while remaining input-dependent, allowing its prediction regions to adapt both their shape and their local geometry to the contextual input.

The multi-norm formulation gives MVCS more geometric flexibility and allows asymmetric prediction regions. However, the multi-norm model introduced by Braun et al. (2026) is not the locally adaptive regression formulation used in their main experiments, since making the multi-norm transformations input-dependent increases the risk of overfitting, as reported in their paper.

The multi-norm variants are still included here as an additional comparison. However, their additional flexibility does not improve the overall efficiency on these regression tasks. The $q \geq 1$ and unrestricted multi-norm variants obtain mean ranks of 6.58 and 6.53, compared with 3.92 and 3.75 for the corresponding single-norm variants, 2.93 for *CoCoNet*$_{\text{Spline-}\ell_1}$, and 1.43 for *CoCoNet*$_{\text{ICNN}}$. These results show that the efficiency advantage of *CoCoNet*$_{\text{ICNN}}$ is not caused only by restricting MVCS to $q \geq 1$. As also shown in Figure 12, the rank of *CoCoNet*$_{\text{ICNN}}$ remains highly competitive against all MVCS variants, while the multi-norm variants result in worse overall ranking and efficiency. This further shows that increasing geometric flexibility alone does not necessarily lead to more efficient prediction regions, and that the combination of flexible geometry and input-dependent adaptation in *CoCoNet*$_{\text{ICNN}}$ provides an important advantage.

Comparison with Flexible Non-Convex Methods. Together with the discussed MVCS variants, the study also includes Pullback Conformal Sets (PBS) Kondratyev et al. (2025). PBS is included as a flexible non-convex baseline because it provides a close comparison to *CoCoNet* among recent multivariate conformal

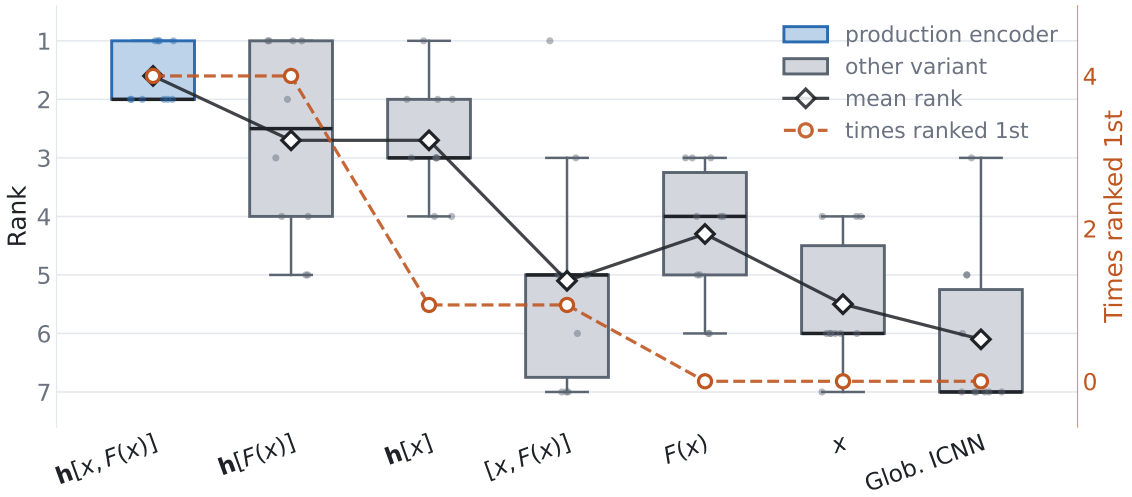

Figure 13: **Efficiency-rank distribution for the *CoCoNet*$_{\text{ICNN}}$ contextual-ablation variants.** Lower rank indicates smaller dimension-normalized prediction-region size. The comparison isolates the effects of the contextual encoder, the information provided to it, and input-dependent versus global scoring. $\mathbf{h}[\cdot]$ denotes the contextual encoder applied to the listed input; the other variants pass that input to the score model directly, and Glob. ICNN uses a single input-independent score. Diamonds denote mean rank, and the dashed curve reports the number of datasets on which each configuration ranks first.

methods. It learns input-dependent joint prediction-region geometry through neural optimal transport and convex neural potentials, while it does not impose convexity on the resulting prediction regions. This provides an additional comparison against a recent flexible non-convex method and allows the efficiency of *CoCoNet* to be evaluated beyond the convex baselines. Therefore, *CoCoNet*$_{\text{ICNN}}$ is compared against a broader set of recent and strong multivariate conformal baselines.

Table 8 shows the overall comparison of *CoCoNet* vs the mentioned basaelines, while PBS is competetive on several datasets It achieves the smallest dimension-normalized size on OES10 and the second-smallest size on ENB and ATP1D. However, it performance varies across datasets with a mean efficiency rank of 5.47 vs 1.30 of *CoCoNet*$_{\text{ICNN}}$

These results provide strong evidence that the advantage of *CoCoNet* is not limited to predefined convex geometric families. It remains highly efficient compared with a flexible transport-based multivariate method, while still producing convex prediction regions by construction. This provides an additional benefit of *CoCoNet*: besides achieving strong efficiency, it preserves convex uncertainty sets, which are particularly useful for downstream decision-making and optimization.

E.1 Ablation Studies

The ablation study evaluates three design choices in *CoCoNet*$_{\text{ICNN}}$: the benefit of learning an input-dependent conformity score, the effect of providing different contextual inputs to the model, and the benefit of using an explicit contextual encoder. Ten representative multivariate datasets are considered, with response dimensions ranging from $p = 2$ to $p = 16$ and covering different sample sizes and application areas.

Table 10 compares seven configurations corresponding to these three design choices. First, the Global *CoCoNet*$_{\text{ICNN}}$ evaluates the benefit of input-dependent score learning by using a single unconditional conformity score for all inputs. Second, three configurations directly condition the ICNN on different contextual inputs: both the input covariates and point prediction $[\mathbf{x}, F(\mathbf{x})]$, only the point prediction $F(\mathbf{x})$, or only the input covariates $\mathbf{x}$. Finally, three additional configurations use the contextual encoder $\mathbf{h}[\cdot]$ with the same three input choices, allowing the effect of explicitly encoding the contextual information before conditioning the ICNN to be evaluated.

Table 10: **Contextual-encoder ablation for *CoCoNet*$_{\text{ICNN}}$ at 90% target coverage.** The comparison isolates the effects of input dependence, the contextual encoder, and the information supplied to the score model. $\mathbf{h}[\cdot]$ denotes the contextual encoder applied to the listed input; the columns without $\mathbf{h}$ pass that input to the score model directly, and Global uses a single input-independent *CoCoNet*$_{\text{ICNN}}$ score. Each configuration reports empirical test coverage in gray followed by dimension-normalized prediction-region size (lower is better); subscripts denote standard deviations over five seeds. Where a factor ($\times 10^k$) follows the dataset name, reported sizes are scaled by 10^k. Best and second-best sizes are highlighted in **blue** and by underlining, respectively.

Dataset	p	$\mathbf{h}[x, F(x)]$		$\mathbf{h}[F(x)]$		$\mathbf{h}[x]$		$[x, F(x)]$		$F(x)$		x		Global	
		Cov. %	Size	Cov. %	Size	Cov. %	Size	Cov. %	Size	Cov. %	Size	Cov. %	Size	Cov. %	Size
ENB	2	$89.4_{\pm 3.5}$	$\underline{7.55_{\pm 0.13}}$	$90.0_{\pm 3.2}$	$9.10_{\pm 1.57}$	$89.6_{\pm 4.4}$	$7.64_{\pm 0.29}$	$92.5_{\pm 2.5}$	$\mathbf{7.40_{\pm 0.59}}$	$90.5_{\pm 5.4}$	$9.53_{\pm 1.02}$	$92.6_{\pm 3.9}$	$8.14_{\pm 0.81}$	$87.5_{\pm 4.1}$	$11.05_{\pm 1.86}$
Bias-Corr. Temp.	2	$90.4_{\pm 0.5}$	$\mathbf{3.88_{\pm 0.13}}$	$89.9_{\pm 1.4}$	$4.03_{\pm 0.10}$	$90.5_{\pm 1.0}$	$\underline{3.89_{\pm 0.10}}$	$90.5_{\pm 0.9}$	$\underline{3.89_{\pm 0.09}}$	$90.2_{\pm 1.0}$	$4.05_{\pm 0.05}$	$90.3_{\pm 0.8}$	$\underline{3.89_{\pm 0.12}}$	$90.1_{\pm 1.3}$	$4.15_{\pm 0.07}$
Online Retail II ($\times 10^1$)	2	$91.2_{\pm 1.1}$	$\underline{2.02_{\pm 0.29}}$	$90.9_{\pm 0.8}$	$\underline{2.02_{\pm 0.18}}$	$90.7_{\pm 0.8}$	$\mathbf{1.93_{\pm 0.19}}$	$90.9_{\pm 0.8}$	$2.33_{\pm 0.16}$	$90.9_{\pm 0.6}$	$2.18_{\pm 0.14}$	$91.0_{\pm 0.6}$	$2.36_{\pm 0.20}$	$90.6_{\pm 1.0}$	$2.81_{\pm 0.96}$
SCPF	3	$89.0_{\pm 3.2}$	$\mathbf{5.16_{\pm 0.68}}$	$91.3_{\pm 2.2}$	$\mathbf{5.16_{\pm 0.40}}$	$90.4_{\pm 2.4}$	$\underline{5.84_{\pm 0.87}}$	$90.8_{\pm 1.9}$	$7.12_{\pm 1.26}$	$91.1_{\pm 1.6}$	$5.93_{\pm 0.61}$	$90.9_{\pm 2.2}$	$7.64_{\pm 1.37}$	$89.5_{\pm 3.2}$	$7.21_{\pm 0.64}$
QM8 ($\times 10^{-1}$)	4	$90.6_{\pm 1.1}$	$\underline{1.10_{\pm 0.03}}$	$90.2_{\pm 0.7}$	$\mathbf{1.08_{\pm 0.03}}$	$90.3_{\pm 0.6}$	$1.19_{\pm 0.04}$	$90.3_{\pm 0.7}$	$1.21_{\pm 0.02}$	$89.9_{\pm 0.9}$	$1.17_{\pm 0.09}$	$90.2_{\pm 0.5}$	$1.26_{\pm 0.02}$	$90.0_{\pm 1.1}$	$1.44_{\pm 0.04}$
Communities Crime ($\times 10^2$)	6	$90.6_{\pm 1.7}$	$\underline{5.94_{\pm 0.17}}$	$91.5_{\pm 2.2}$	$\mathbf{5.70_{\pm 0.18}}$	$91.6_{\pm 1.2}$	$6.07_{\pm 0.07}$	$91.6_{\pm 1.7}$	$7.37_{\pm 0.80}$	$90.4_{\pm 1.5}$	$6.54_{\pm 0.56}$	$91.8_{\pm 1.4}$	$7.26_{\pm 0.73}$	$91.6_{\pm 1.8}$	$6.72_{\pm 0.67}$
QM9-8D	8	$90.0_{\pm 0.9}$	$\mathbf{4.94_{\pm 0.08}}$	$90.5_{\pm 0.8}$	$5.02_{\pm 0.10}$	$90.3_{\pm 0.9}$	$\underline{4.99_{\pm 0.08}}$	$90.2_{\pm 0.7}$	$5.11_{\pm 0.12}$	$89.8_{\pm 0.9}$	$5.02_{\pm 0.14}$	$90.3_{\pm 0.8}$	$5.16_{\pm 0.10}$	$90.4_{\pm 0.9}$	$5.57_{\pm 0.27}$
Knee Contact Force ($\times 10^1$)	12	$90.3_{\pm 2.3}$	$\underline{6.65_{\pm 0.33}}$	$90.0_{\pm 1.7}$	$\mathbf{6.64_{\pm 0.23}}$	$90.0_{\pm 2.0}$	$6.76_{\pm 0.40}$	$89.9_{\pm 2.0}$	$7.31_{\pm 0.62}$	$90.5_{\pm 1.3}$	$6.91_{\pm 0.25}$	$89.7_{\pm 1.6}$	$6.77_{\pm 0.11}$	$90.0_{\pm 1.8}$	$7.60_{\pm 0.39}$
QM7b	14	$90.1_{\pm 0.6}$	$\mathbf{1.78_{\pm 0.03}}$	$91.0_{\pm 0.9}$	$1.93_{\pm 0.12}$	$90.5_{\pm 1.1}$	$\underline{1.84_{\pm 0.03}}$	$90.2_{\pm 0.7}$	$3.17_{\pm 0.65}$	$90.2_{\pm 0.8}$	$1.89_{\pm 0.05}$	$90.4_{\pm 0.6}$	$2.49_{\pm 0.34}$	$89.8_{\pm 0.6}$	$2.10_{\pm 0.05}$
OES97 ($\times 10^3$)	16	$92.7_{\pm 2.2}$	$\underline{2.87_{\pm 0.29}}$	$90.0_{\pm 5.6}$	$\mathbf{2.79_{\pm 0.34}}$	$91.8_{\pm 3.9}$	$3.56_{\pm 0.43}$	$91.8_{\pm 5.2}$	$20.51_{\pm 7.39}$	$93.3_{\pm 3.0}$	$3.87_{\pm 0.52}$	$91.5_{\pm 4.3}$	$18.35_{\pm 5.40}$	$90.3_{\pm 4.9}$	$2.94_{\pm 0.36}$
Mean rank ↓		$\mathbf{1.60 \pm 0.49}$		$\underline{2.70 \pm 1.62}$		$\underline{2.70 \pm 0.90}$		5.10 ± 1.81		4.30 ± 1.10		5.50 ± 1.02		6.10 ± 1.30	

Note. Large datasets are randomly subsampled to 10,000 *observations for the ablation study, with the same sampled data used across all variants within each seed.*

Figure 13 summarizes the efficiency ranks of the seven configurations. As shown in the figure, the Global *CoCoNet*$_{\text{ICNN}}$ has the worst overall performance, with a mean efficiency rank of 6.10, showing the importance of making the conformity score input-dependent. Providing contextual inputs directly to the ICNN improves the efficiency across the next three configurations, supporting the benefit of conditioning the score on the input covariates, the point prediction, or both. The results further improve when the same contextual inputs are first processed through the encoder. Among all seven configurations, using the encoder with both the input covariates $\mathbf{x}$ and the point prediction $F(\mathbf{x})$ achieves the best overall performance and the lowest mean efficiency rank. This configuration also shows more stable performance across the selected datasets and consistently remains among the top-performing variants, supporting the final design of *CoCoNet*.