

---

# Adaptive Conformal Regression with Jackknife+ Rescaled Scores

---

**Nicolas Deutschmann**  
IBM Research  
[deu@zurich.ibm.com](mailto:deu@zurich.ibm.com)

**Mattia Rigotti**  
IBM Research  
[mrg@zurich.ibm.com](mailto:mrg@zurich.ibm.com)

**María Rodríguez Martínez**  
IBM Research  
[mrm@zurich.ibm.com](mailto:mrm@zurich.ibm.com)

## Abstract

Conformal regression provides prediction intervals with global coverage guarantees, but often fails to capture local error distributions, leading to non-homogeneous coverage. We address this with a new adaptive method based on rescaling conformal scores with an estimate of local score distribution, inspired by the Jackknife+ method, which enables the use of calibration data in conformal scores without breaking calibration-test exchangeability. Our approach ensures formal global coverage guarantees and is supported by new theoretical results on local coverage, including an *a posteriori* bound on any calibration score. The strength of our approach lies in achieving local coverage without sacrificing calibration set size, improving the applicability of conformal prediction intervals in various settings. As a result, our method provides prediction intervals that outperform previous methods, particularly in the low-data regime, making it especially relevant for real-world applications such as healthcare and biomedical domains where uncertainty needs to be quantified accurately despite low sample data.

## 1 Introduction

Conformal prediction (CP) [1–3] provide a framework to perform rigorous post-hoc uncertainty quantification on machine learning predictions. CP converts the predictions of a model into sets of predictions that can guarantee any desired expected coverage (the probability that the right answer is contained in the predicted set) with finite calibration data and no constraints on distributions.

There has been much recent work developing the original idea of CP to make it more computationally efficient [4, 5], generalize the hypotheses of the method [6, 7], control Type I and Type II errors [8, 9], and ensure different types of conditional validity [10, 11]. The original formulation of CP was best-suited for classification problems, and indeed the early work of Lei and Wasserman [12] identified that a straightforward application of the formalism would predict constant-size intervals as prediction sets, highlighting the need for methods to make these prediction intervals (PI) dynamic. The establishment of a theoretical basis for conformal regression (CR) by Lei et al. [13] was followed by multiple new approaches for CR that maintain the original guarantees of CP while also improving local behavior [14–18].

These approaches take three possible routes: early work focused on devising non-conformity scores that themselves capture some locally-dependent information [19, 13], while a second wave of methods relied on non-dynamic CP applied on machine learning (ML) models that predict sets or intervals [14, 20]. Finally, the seminal work of Guan [15] on dynamically weighting the nonconformity score empirical distribution sparked a healthy influx of new methods [18, 16, 17, 21].

Our work revisits the early approach of encoding local knowledge in the score function itself, in light of the general advances of the field of conformal predictions. In particular, we consider the locally-rescaled absolute error score of Lei et al. [13] (MADSPLIT), which was formulated in the

context of split-conformal predictions and therefore relies on the training error score distribution, often very different from test errors due to overfitting. We propose a new approach to data splitting, combining both split-conformal and Jackknife+ [22], a recent method that modifies the jackknife to achieve prediction coverage guarantees. This allows predictions with a single ML model, local error scale measures on the calibration data, while preserving coverage guarantees. Our method does not have formal *local* coverage guarantees, but we provide new bounds on local coverage based on the statistical dependence of the input data and the nonconformity scores which allow us to tune our score function to obtain close-to-optimal prediction intervals. We complement these theoretical results with detailed empirical comparison with existing methods on numerical, sequence, and image data, showing the benefits of our solution both in terms of data efficiency and tuning.

### 1.1 Contributions

- We introduce a new method to rescale nonconformity scores by an *in-distribution* local mean estimate, using a combination of the split-conformal and Jackknife+ schemes.
- We prove a global marginal coverage guarantee for our new approach.
- We prove a new bound on local coverage based on the mutual information between the non-conformity scores and the input data.
- We show how this bound combined to our scheme enables a new principled tuning of score localizers, leading to improved local coverage.
- We compare our method empirically with other approaches, and show that our method is more robust to low calibration statistics than approaches based on reweighted empirical score distributions, while avoiding the biases of the original MADSPLIT approach to rescaled scores.

### 1.2 Related Work

There are three main classes of approaches for providing adaptive prediction intervals (PI) for conformal regression.

**Reweighted scores.** The earliest efforts on adaptive CR proposes to rescale the usual nonconformity scores [19, 13] with local information. The most prominent approach is the MADSPLIT approach [13], which uses  $s(\hat{y}(X), y) = |\hat{y}(X) - y|/\hat{\sigma}(X)$ , where  $\hat{\sigma}(X)$  is an estimate of the conditional score mean  $\mathbb{E}[|\hat{y}(X) - y||X]$ . Importantly in comparison with our proposed method, in MADSPLIT this estimate is performed on the training split, which preserves CP coverage guarantees, but yields poor performance when errors differ between training and test sets.

**Model-based interval proposals.** A second type of approach is based on choosing machine-learning models that themselves encode some notion of uncertainty. The canonical formulation is conformal quantile regression [14], although other approaches based on predicting distribution parameters exist. These are applicable provided that the conditional label distribution  $p(y|X)$  is slow-varying and wide enough that predicting conditional quantiles is a valid learning objective. Another limitation of these methods is that poor modelling performance immediately leads to poor PI. This makes PI for difficult-to-predict examples the least trustworthy, despite often being the most important.

**Reweighted score distributions.** The most recent type of adaptive conformal regression methods is based on computing adaptive distortions of the non-conformity score empirical cumulative distribution function (ECDF), proposed originally by Guan [15]. Given a test point, the ECDF is modified by giving a similarity-based weight to each calibration point, yielding an estimate of the test-point-conditional score ECDF. If the ECDF estimator is good, the result is perfect local coverage and adaptivity. Multiple variants have been proposed to optimize computational complexity or ECDF estimation, including LCR [15], LVD [18], SLCP [17]. In any case, estimating the high quantiles of the conditional CDF can require significant statistics, as many calibration points are needed *near every test point*. Furthermore, as we discuss in section 4, the proposed localization-weight tuning methods often yield poor results on complex, high-dimensional data.

### 1.3 Definitions

Let us establish definitions that will be reused throughout this manuscript. We consider a probability space  $M_{X \times y} = (\Omega_X \times \Omega_y, \mathcal{F}_X \otimes \mathcal{F}_y, \pi)$  defined over a product of measurable spaces<sup>1</sup>,  $(\Omega_X, \mathcal{F}_X)$ , and  $(\Omega_y, \mathcal{F}_y)$ , to which we refer respectively as *input-* and *label-space*.

We work in the context of split-conformal predictions, *i.e.*, we sample a training dataset which we use to produce a predictive model  $m(X) : \Omega_X \rightarrow \Omega_y$  through a training algorithm. We then further sample *i.i.d.*  $(X_i, y_i)_{i=1 \dots N+1}$ , also independent from the training data and  $m$ . We refer to the first  $N$  sampled points as the calibration dataset  $\mathcal{C}_N = (X_i, y_i)_{i=1 \dots N}$  and to  $(X_{N+1}, y_{N+1})$  as the test point. For conformal predictions, we consider a score function  $s(X, y) : \Omega_X \times \Omega_y \rightarrow \mathbb{R}$  and define the usual calibration intervals with marginal risk  $\alpha \in [0, 1]$  as  $S^\alpha(X) = \{y \in \Omega_y | s(X, y) \leq q_{\mathcal{C}_N}^{1-\alpha}(s)\}$ , where  $q_{\mathcal{C}_N}^{1-\alpha}(s)$  is the  $\lceil(1 - \alpha)(n + 1)\rceil/n$ -quantile of the empirical score distribution on  $\mathcal{C}_N$ .

## 2 Local Coverage and Score-Input Independence

We begin by defining and characterizing the goals of adaptive conformal regression in terms of guarantees and tightness. This analysis is general and extends beyond the method we propose, but will also provide motivation for our approach and a framework to understand its applicability.

Informally, the objective of PI prediction to achieve two desirable properties: local coverage guarantees for any input point  $X$  and interval bounds that perfectly adapt to capture the label distribution.

### 2.1 Local Coverage Guarantees

We propose a slightly extended notion of conditional coverage compared to that of Han et al. [17], which is more suitable to proving the bound of proposition 2.

**Definition 1** (Input-space strong conditional coverage (ISCC))

We define  $\alpha$ -input-space strong conditional coverage as the following property:

$$\forall \omega_X \in \mathcal{F}_X, \mathbb{P}_{X, y \sim \pi} \left( y \in S^{(\alpha)}(X) \mid X \in \omega_X \right) \geq 1 - \alpha.$$

The formulation of definition 1 strongly hints at as sufficient condition to ensure strong conditional coverage for conformal intervals defined as in section 1.3. Weaker conditional coverage properties can be defined as the same condition applying to specific subsets of  $\mathcal{F}_X$ .

**Proposition 1** (Sufficient condition for ISCC)

If  $S^{(\alpha)}(X)$  is defined from a score  $s(X, y)$ ,

$$X \perp s(X, y) \Rightarrow \alpha\text{-ISCC}$$

Indeed, if the score distribution is input-independent, then its conditional quantiles are as well, so that estimating quantiles globally is the same as estimating them locally. This has been observed in previous work and is tightly related to the definition of the orthogonal loss in Feldman et al. [20], which measures the correlation between the local coverage and the local interval size.

Score-input independence is never realized in practice, and we are not aware of any coverage bound based on the orthogonal loss when optimality is not achieved. As we show below, however, mutual information can be used instead to place a bound on local coverage.

**Proposition 2** (Bound on conditional coverage)

If the mutual information between  $X$  and  $s(X, y)$ ,  $\text{MI}(X, s)$  is finite, for any  $\omega_X \in \mathcal{F}_X$  such that  $\mathbb{P}(X \in \omega_X) > \rho$

$$\mathbb{P}(y \in S^\alpha(X) | X \in \omega_X) \geq (1 - \tilde{\alpha}), \text{ where } \tilde{\alpha} = \alpha + \frac{\sqrt{1 - e^{-\text{MI}(X, s)}}}{\rho}.$$

---

<sup>1</sup>It goes without saying that the probability distribution  $\pi$  is not a product measure, *i.e.*  $y \not\perp X$ .

Note that this bound applies the coverage probability marginalized over the calibration data. The actual coverage given a calibration dataset is a random variable which can deviate below the bound significantly if the calibration sample size is limited [10].

This results is a direct consequence of proposition 2, and is proven in the supplementary material. The bound is vacuous for low-probability sets, which, as we argue in appendix S.6.1, is likely inherent to this type of result. On the other hand, given a score function with free parameters, one can extend its usefulness by optimizing for smaller score-input mutual information, as we illustrate in appendix S.3.2.

## 2.2 Measuring the Adaptivity of PI

It is common to evaluate adaptive CR methods based on two metrics computed on held-out test data: the coverage rate (Cov.) and the mean PI size (IS), under the understanding that, at fixed coverage, smaller intervals means tighter correlation between PI size and error rate. However, as we show in appendix S.4, an oracle predicting ideal intervals might actually mean *increasing* the mean interval size compared to a non-adaptive approach. We therefore propose several new metrics to evaluate the adaptivity of a given PI prediction method, using absolute errors as our conformal score (CS):

- $\tau_{SI}$ , the point-wise Kendall rank correlation coefficient between the CS and the PI size (IS)
- $\tau_{SQI}$ , the Kendall correlation between IS quantiles (ISQ, we usually use deciles) and the ISQ-conditional score  $(1 - \alpha)^{\text{th}}$  quantile (CSQ)
- $R_{SQI}^2$ , the  $R^2$  regression coefficient for the linear model  $\text{ISQ} = 2 \times \text{CSQ}$ , where we use the middle point of each IS quantile bin as value for  $\text{ISQ}^2$

When the score is the absolute error,  $\text{IS} = 2 \times q^{(1-\alpha)}(\text{CS}|IS)$  is a necessary condition for reaching the best possible interval-local error coupling. We propose  $R_{SQI}^2$  to measure the validity of this relation, which relies on discretizing the interval size distribution to measure conditional error quantiles.

While nearly-perfect solutions can be compared with  $R_{SQI}^2$ , it treats adaptive but over-conservative PI on the same footing as PI with no adaptivity. We therefore also use the rank correlation coefficient  $\tau$  as a measure of monotonicity, using point-level metrics in  $\tau_{SI}$ , and quantile-aggregated metrics in  $\tau_{SQI}$ , which is less sensitive to distribution shape details. Informally, these two metrics can be thought of as framing the orthogonal loss of Feldman et al. [20] in terms of how stringent the score-interval size relationship is constrained. Our  $\tau_{SQI}$  can also be seen as a different compromise in granularity: we recover continuous information about coverage by using the conditional score quantiles but discretize the interval sizes.

## 3 Theoretical Results: Jackknife+ Rescaled Conformal Scores

### 3.1 Locally Rescaled Conformity Metric

Considering the observations made in section 2, one can note that, for a given  $\alpha$ , we can achieve ISCC by rescaling score function  $s$ :

$$\sigma_{q_\alpha}(X, y) = \frac{s(X, y)}{q^{(1-\alpha)}(s|X)}, \quad (1)$$

where  $q^{(1-\alpha)}(s|X)$  is the  $(1 - \alpha)$ -th quantile of the conditional score distribution  $p(s(X, y)|X)$ , which is essentially what ECDF-based methods aim to do. It is however costly to estimate these conditional quantiles and we propose to revisit the generalisation of the rescale absolute error used in MADSPLIT [13]

$$\sigma(X, y) = \frac{s(X, y)}{\hat{s}(X)}, \quad (2)$$

---

<sup>2</sup>Note that Kendall's  $\tau$  is rank-based and therefore the rank of each quantile can be used instead of a numerical value for  $\tau_{SQI}$  and  $\tau_{SAI}$ .

where  $\hat{s}(X)$  is an estimator of the mean of  $s$  conditioned on  $X$ ,  $\bar{s}(X)$ . It is easy to show that this score yields again the perfect results of eq. (1) when the random variable  $s(X, y)|X$  is a scale family indexed by  $X$ . Empirically, it is also often the case that this ratio has reduced  $X$ -dependence compared to the raw score  $s$ .

The conditional mean can be estimated with a Nadaraya–Watson estimator based on a kernel  $K$ :

$$\hat{s}_i = \sum_{j=1; j \neq i}^N p_{ij}^{(K)} s_j, \quad \text{where } p_{ij}^{(K)} = \frac{K(X_i, X_j)}{\sum_{k \leq N; k \neq i} K(X_i, X_k)}. \quad (3)$$

Formal guarantees on the convergence of this estimator typically require fine-grained knowledge of the data distribution to be made quantitative, which is rarely the case in practice. Nevertheless, proposition 2 provides a tool to guide kernel choice and tuning by minimizing  $\text{MI}(X, s)$ .

In MADSPLIT,  $\hat{s}_i$  is estimated on the training data, which is risky due to the training-test error distribution shift, especially for modern deep learning [23]. If we were to instead use the calibration data as in eq. (3), we would predict the following test interval

$$C_{N+1} = \mu(X_{N+1}) \pm \hat{s}_{N+1} q^\alpha (\{\sigma_i\}_{i \in \text{cal}}). \quad (4)$$

This, however, breaks the exchangeability of the test point with the calibration set, which breaks the hypotheses of split-CP. Further splitting the calibration dataset is another possible option, but this requires sacrificing the variance of the calibration-conditional coverage [10]. Our new method described below uses a modification of eq. (3) to restore formal guarantees.

### 3.2 Exchangeable Rescaled Scores with Jackknife+

We propose an approach that combines elements of the split conformal and Jackknife+ approaches [22] by splitting the data into a training and calibration set, and applying the Jackknife+ on the calibration set to train an estimator of the conditional conformal score mean.

After training a model, we consider a matrix of scores, indexed by the union of the calibration set and the test point,  $J^+ = [1 \dots N + 1]$ . Each score itself is evaluated by computing every estimator over a restricted set  $J_{ij}^+ = J^+ / \{i, j\}$ :

$$\hat{s}_{ij} = \sum_{k \in J_{ij}^+} p_{ik;j}^{(K)} s_k, \quad \text{where } p_{ik;j}^{(K)} = \frac{K_{ij}(X_i, X_j)}{\sum_{l \in J_{ij}^+} K_{ij}(X_i, X_l)}. \quad (5)$$

We introduced the kernels  $K_{ij}$  to allow kernel tuning: all  $K_{ij}$  can be taken from the same family of kernels and have their parameters optimized to minimize the test-score mutual information  $\text{MI}(s, X)$  estimated on  $J_{ij}^+$ . Another option is to fix a kernel *a-priori*, which the robustness of the mean-rescaled score easily allows.

We can express the scores as  $s_{ij}^+ = \frac{|\mu(X_i) - Y_i|}{\hat{s}_{ij}}$ , and in particular the actual calibration scores are  $s_i^+ = s_{i(N+1)}^+$ . These  $(N + 1) \times (N + 1)$  scores are truly exchangeable over the whole calibration set extended with the test point. There are now  $N$  scores for the test point, which get folded into the interval definition as follows:

$$C_{N+1}^{+\alpha} = [\mu_{X_{N+1}} - q^\alpha (\{\hat{s}_{N+1,i} s_i^+\}), \mu_{X_{N+1}} + q^\alpha (\{\hat{s}_{N+1,i} s_i^+\})]. \quad (6)$$

We provide a summary of our procedure in algorithm S.1

### 3.3 Global Coverage Guarantee

When using the conformal prediction intervals defined in eq. (6), we obtain exactly the same type of coverage guarantee as the original Jackknife+ approach provides:

**Proposition 3** (Global Coverage of  $C_{N+1}^{+\alpha}$ )

*Marginalizing over the calibration set:*

$$\mathbb{P}(Y_{N+1} \in C_{N+1}^{+\alpha}(X_{N+1})) \geq 1 - 2\alpha.$$

This coverage is degraded compared to the coverage probability of standard conformal predictions. Nevertheless, as for the original Jackknife+ approach, we observe empirically that the effective coverage is most often close to  $(1 - \alpha)$ . As for theorem S.2, one should keep in mind that this bound is marginalized over calibration data.

## 4 Experimental Evaluation

Throughout this section, unless mentioned otherwise, we do not use the kernel tuning capabilities of our approach, and fix the kernel to a simple (approximate [24, 25]) K-nearest-neighbor (KNN) kernel ( $K_{ij} = 1$  if  $j$  is among the KNN of  $i$ , otherwise  $K_{ij} = 0$ ), setting  $K = 10$ . We've found that this approach yields competitive results in many cases without any tuning for our approach. We discuss the potential gains of kernel tuning in appendix S.3.2.

### 4.1 Comparing Fairly to Previous Work

Tuning a localizer is a key element of all existing post-hoc adaptive PI and should, to some extent, be considered part of the method. In particular, our proposal is, to our knowledge, the only one using training-independent score information without sacrificing  $\mathcal{O}(N_{\text{cal}})$  data, and cannot be applied to the baselines to which we compare. To perform fair comparisons, we have tried to strike a balance between considering PI proposal methods as end-to-end and evaluating the CP method in isolation by making best effort extensions.

We evaluate our method against MADSPLIT [13] and LVD [18] as representatives of the two classes of post-hoc CR. We have indeed found that their proposed methods for kernel tuning lead to poor performance in our experiments, which we report in appendix S.3. We use modified kernels that yield better performance and therefore help assess the CR methods themselves. For MADSPLIT, we use the same KNN kernel as for our method, but these are unsuited for LVD. The original kernel for LVD is an anisotropic RBF ( $K(x, y) = e^{-|A(x-y)|^2}$ ) optimized on training data. We found that the matrix  $A$  is usually too large, yielding mostly infinite intervals. We therefore replace it by  $\lambda A$ , tuned on subsampled training data to have at least  $(1 - \alpha)$  samples with effective sample size<sup>3</sup> at least  $2/\alpha$ , allowing mostly finite intervals. For metrics, we replace infinite interval sizes by the largest observed calibration error. Note that we also attempted to evaluate SLCP [17], but were unable to tune kernels to reach comparable metrics to the other methods, and we therefore left it out of our results.

### 4.2 One-Dimensional Regression

Let us start by applying our methods to simple regression problems that will help illustrate how our approach operates. We define the following random variables

$$\begin{aligned} X &\sim U(0, 1), & y &= f(X) + \epsilon, & \epsilon &\sim \mathcal{N}(0, \nu(X)), \\ f(X) &= f_0 + X^2 \sin(k_f X + \phi_f), & \nu(X) &= \epsilon_0 + |\sin(k_\epsilon X + \phi_\epsilon)| \end{aligned} \tag{7}$$

We use 1000 calibration and 10000 test points to evaluate our method on a random-forest regressor trained on independent data and indeed find intervals that achieve the target global coverage and dynamic interval sizes, as shown in fig. 1.

This simple setting already allows us to compare performance with other adaptive PI methods. We display the calibration-set-size dependence of PI metrics in fig. 2.

These results highlight the case we make more systematically below: in the low data regime, our method is more robust than ECDF-based methods such as LVD due to its better data efficiency, and

---

<sup>3</sup>Measured as  $\exp H(w)$  where  $w$  is the kernel-reweighted MDF on the calibration set.

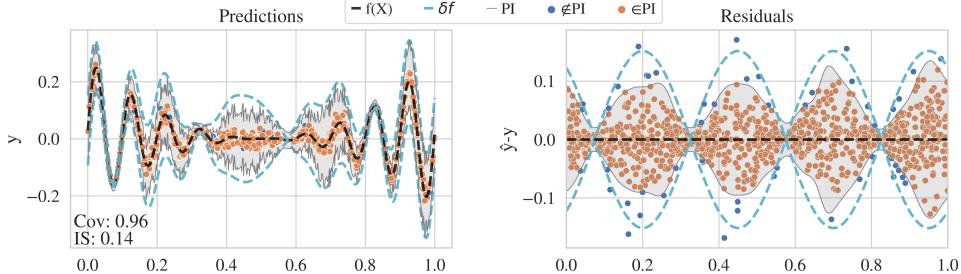


Figure 1: Evaluation of our approach on a 1D regression problem.

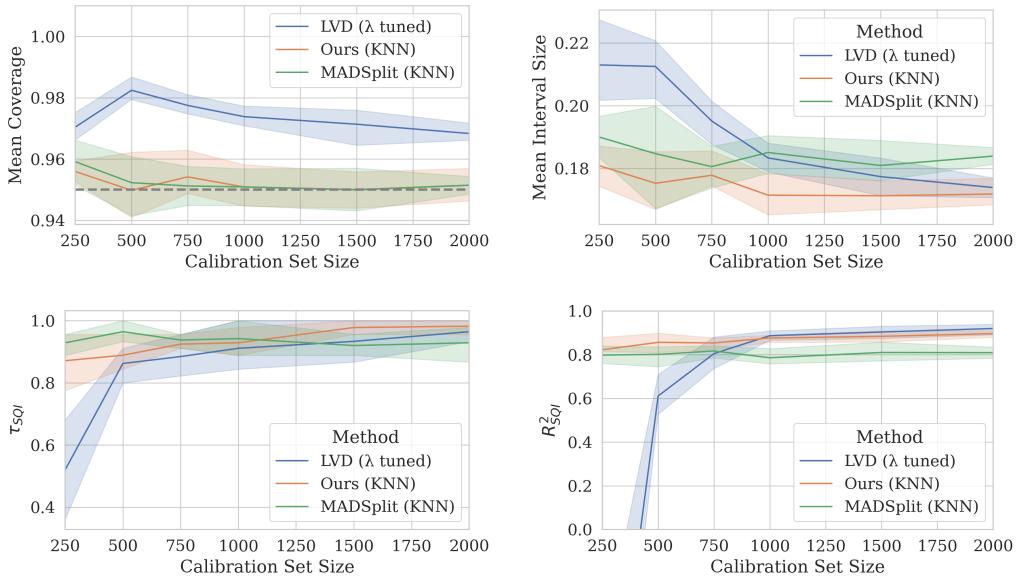


Figure 2: Calibration-set-size dependence of PI metrics evaluated on an independent test set.

has comparable performance with them when more data is available. In this example, there is no significant difference with MADSPLIT in the low-data regime, while we outperform it in the high data regime, which we attribute to the training-test error distribution shift.

#### 4.2.1 Results on complex data

In order to assess and compare the performance of our approach and previous work, we selected a number of regression task spanning data modalities and model performances. We provide results in the high- and low-calibration data regimes and compute the metrics defined in section 2.2. All experiments are performed with a target coverage of  $1 - \alpha = 95\%$ , repeating over 10 splits of the held-out data between calibration and test sets.

We consider four datasets in addition to our 1D regression problem. In all cases, we fix the training data and train an adapted regression model by optimizing the mean squared error. Details about data processing, embeddings used for localisation and predictive models can be found in appendix S.2. The tasks on which we evaluate are:

**Two TDC molecule property datasets [26]:** drug solubility and clearance prediction from respectively 4.2k and 1.1k SMILES sequences<sup>4</sup>.

**The ALPHASEQ Antibody binding dataset [28]:** binding score prediction for a SARS-CoV-2 target peptide from 71k amino-acid sequences.

<sup>4</sup>string-based description of molecular structures [27].

Dataset	$N_{\text{cal}}$	Method	Cov. ( $\gtrsim 0.95$ )	IS $\downarrow$	$R_{SQI}^2 \uparrow$	$\tau(SQI) \uparrow$	$\tau(SI) \uparrow$
1D	500	Ours	0.96(1)	<b>0.18(1)</b>	<b>0.87(3)</b>	0.91(4)	<b>0.35(1)</b>
		MADSPLIT	0.952(9)	<b>0.18(1)</b>	0.80(4)	<b>0.96(3)</b>	<b>0.344(9)</b>
		LVD	0.982(3)	0.21(1)	0.61(9)	0.86(4)	0.33(1)
	2000	Ours	0.952(2)	<b>0.171(5)</b>	<b>0.90(2)</b>	0.96(4)	<b>0.36(1)</b>
		MADSPLIT	0.952(9)	0.18(1)	0.80(4)	0.96(3)	0.344(9)
		LVD	0.968(3)	<b>0.174(3)</b>	<b>0.92(2)</b>	0.96(3)	<b>0.361(6)</b>
TDC Sol.	400	Ours	0.95(1)	5.3(5)	0.4(1)	0.6(1)	0.09(1)
		MADSPLIT	0.95(1)	5.0(3)	<b>0.53(5)</b>	<b>0.71(6)</b>	<b>0.182(3)</b>
		LVD	0.958(7)	5.2(3)	0.3(5)	<b>0.7(1)</b>	0.07(5)
	800	Ours	0.96(1)	<b>5.2(4)</b>	0.5(1)	0.7(1)	0.11(1)
		MADSPLIT	0.951(6)	<b>5.1(2)</b>	<b>0.58(7)</b>	0.71(5)	<b>0.179(6)</b>
		LVD	0.960(9)	5.4(2)	0.4(3)	0.77(7)	0.07(1)
TDC Clear.	60	Ours	0.95(2)	<b><math>1.9(3) \cdot 10^2</math></b>	<b>0.2(3)</b>	<b>0.6(1)</b>	<b>0.37(3)</b>
		MADSPLIT	0.95(2)	<b><math>1.9(5) \cdot 10^2</math></b>	<b>0.2(3)</b>	<b>0.60(9)</b>	<b>0.39(1)</b>
		LVD	0.97(2)	$2.2(2) \cdot 10^2$	$\ll 0$	-0.3(6)	0.1(2)
	240	Ours	0.96(1)	<b><math>1.65(9) \cdot 10^2</math></b>	<b>0.3(2)</b>	<b>0.5(1)</b>	<b>0.39(7)</b>
		MADSPLIT	0.96(1)	<b><math>1.64(8) \cdot 10^2</math></b>	<b>0.41(5)</b>	<b>0.6(1)</b>	<b>0.41(7)</b>
		LVD	0.97(1)	$2.02(5) \cdot 10^2$	$\ll 0$	0.1(2)	0.1(1)
$\alpha$ Seq CoVID	2000	Ours	0.952(6)	<b>4.5(1)</b>	<b>0.24(8)</b>	<b>0.5(1)</b>	<b>0.05(1)</b>
		MADSPLIT	0.952(5)	<b>4.6(1)</b>	<b>0.1(2)</b>	0.2(2)	0.01(2)
		LVD	0.994(1)	7.6(5)	$\ll 0$	0.1(1)	-0.004(6)
	10000	Ours	0.953(4)	<b>4.54(6)</b>	<b>0.28(6)</b>	<b>0.6(1)</b>	<b>0.052(8)</b>
		MADSPLIT	0.952(2)	<b>4.59(6)</b>	0.0(1)	0.2(2)	0.01(1)
		LVD	0.973(4)	5.49(4)	-0.26(4)	0.5(1)	<b>0.052(8)</b>
MNIST	2000	Ours	0.952(6)	<b>0.62(3)</b>	$\ll 0$	<b>0.90(5)</b>	0.20(1)
		MADSPLIT	0.951(7)	<b>0.63(5)</b>	$\ll 0$	0.83(8)	0.17(1)
		LVD	0.988(4)	4.1(5)	<b>-0.19(2)</b>	<b>0.86(5)</b>	<b>0.21(1)</b>
	5000	Ours	0.950(3)	<b>0.601(8)</b>	$\ll 0$	<b>0.92(5)</b>	<b>0.23(1)</b>
		MADSPLIT	0.947(4)	0.62(1)	$\ll 0$	0.85(7)	0.193(6)
		LVD	0.984(2)	4.4(2)	<b>-0.17(3)</b>	<b>0.92(3)</b>	<b>0.233(9)</b>

Table 1: Numbers in parentheses are the uncertainty on the last significant digit evaluated over 10 random samples of test and calibration data. We report  $R_{SQI}^2 \ll 0$  if it is significantly lower than -2.

**Regression-MNIST [29]:** Floating-point prediction of the label of MNIST images with test-time augmentations. This task was selected due to its irregular error distribution, which is expected to challenge our rescaling approach, see appendix S.3.4.

We detail our results in table 1. In summary, we confirm our results that our method outperforms LVD when data is scarce, while we are usually comparable with MADSPLIT, and that our performance improves faster than MADSPLIT with statistics. The results on sequence data with higher statistics highlight confirm that our method is competitive with both method. Interestingly, while AlphaSeq has the most data, our method dominates others. This is due to a combination of increased overfitting and scattered data distribution as discussed in appendix S.3.3. Finally, the MNIST task is peculiar: all methods have comparable performance as measured by correlation measures but only LVD captures anywhere close to the ideal absolute interval size, as measured by  $R_{SQI}^2$ . We attribute this to the irregular error distributions due to possible number confusions, which leads to highly variable conditional mean/quantile ratios, as shown in appendix S.3.4.

## 5 Limitations and Risks

Our analysis shows our method has great promise for extending the applicability of conformal regression to settings where error rates are variable, its success is dependent on a number of factors whose absence can lead to failure.

**Score mean/quantile coupling condition.** The definition of our conformal scores relies on the hypothesis that the conditional score distribution shifts with  $X$  mostly by rescaling. In general, our method’s adaptivity will degrade with increasing variance of  $q_{1-\alpha}(s|X)/\mathbb{E}(s|X)$ , which is exemplified by the results on the MNIST regression task, and discussed in details in appendix S.3.4. This ratio cannot be bounded in general, but turns out to be moderate enough in practice for our approach to work in many cases. Alternative constraints can be derived based on concentration inequalities as shown in appendix S.7, and might be more suitable empirical assessment.

**Non-smooth score-input relationship.** Our approach based on Nadaraya-Watson estimators of local score distribution means rely on having input representations (combining embeddings and kernels) that change smoothly enough that they can be captured. A poor data representation can break the input-score relationship and break adaptivity. This is an issue with all existing adaptive regression methods and rely on valid design choices.

**Risks.** Coverage guarantees are most commonly formulated in terms of coverage probabilities marginalized on the calibration data, and our method is no exception. While clear to experts, potential downstream users might not realize the variability of conditional coverage, which is paramount to understand for system certification. We suggest that marginal coverage guarantees always be followed by a clear warning of the marginal nature of the guarantee. Software making CP should also provide such warnings or bounds. For our approach, we’ve tried to make the current absence of PAC bounds clear, and do so as well in the code to be released with this paper’s final version. Given the simple learning algorithm used in the Jackknife+ part of our approach (NW mean estimator), it is likely that we escape the no-go theorem of Bian and Barber [30] and we hope to show in further work that some version of the PAC bounds established in Barber et al. [22] apply.

## 6 Conclusions

Our results show that our new approach provides a satisfying solution to the weaknesses of existing post-hoc adaptive conformal regression methods. On the one hand, our use of the Jackknife+ procedure to tune and evaluate the conditional mean of the non-conformity score solves the issue of MADSPLIT when evaluated on models with significant score distribution shifts between model-training and CP calibration data. On the other hand, while our method does not guarantee perfect local coverage asymptotically unless stringent hypotheses are verified, our empirical results show clear value compared to the diminished statistical power compared to ECDF-reweighting methods, especially in the low-calibration-data regime.

The lack of satisfying tuning criteria for ECDF-based methods also easily leads to infinite PI, even with abundant data, another point in favor of our approach. Our results indicate that ECDF-based methods tend to have very low tolerance for non-optimal localizers in high-dimensional settings, making further exploration of localizers for ECDF CR critical: these techniques are provably asymptotically optimal adaptive solutions given a good localizer, making them appealing for data-rich domains. Short of that, the comparative robustness of our proposed solution makes it an attractive alternative not only when data is scarce, but also for high-dimensional settings such as deep learning.

It nevertheless remains unclear whether PAC bounds can be established for our method, much like for pre-existing work. This is a crucial element for the applicability of any CP method in safety-critical domains, as guarantees marginalized on calibration data cannot be used to certify error rates in specific implementations. We have good hope that further investigation of the properties of our method, potentially extending it to a CV+ approach to circumvent the no-go theorem of Barber et al. [7], will yield such constraints, further reinforcing our method as a good candidate for uncertainty estimation in regression problems where rigorous uncertainty quantification is essential.

## Acknowledgments and Disclosure of Funding

We thank Jannis Born, Anna Weber and Aurélien Pélissier for useful discussions. This work was supported by the Swiss National Science Foundation Grant No. 192128, and by the European Union’s Horizon 2020 research and innovation programme under grant agreements No. 101070408 (SustainML) and No. 826121 (iPC).

## References

- [1] Volodya Vovk, Alexander Gammerman, and Craig Saunders. Machine-Learning Applications of Algorithmic Randomness. In *Proceedings of the Sixteenth International Conference on Machine Learning*, ICML ’99, pages 444–453, San Francisco, CA, USA, June 1999. Morgan Kaufmann Publishers Inc. ISBN 978-1-55860-612-8. [[↑ page 1](#)]
- [2] C. Saunders, A. Gammerman, and V. Vovk. Transduction with Confidence and Credibility. In *Sixteenth International Joint Conference on Artificial Intelligence (IJCAI ’99) (01/01/99)*, pages 722–726, 1999. URL <https://eprints.soton.ac.uk/258961/>.
- [3] Vladimir Vovk, A. Gammerman, and Glenn Shafer. *Algorithmic Learning in a Random World*. Springer, New York, 2005. ISBN 978-0-387-00152-4 978-0-387-25061-8. [[↑ page 1](#)]
- [4] Harris Papadopoulos, Kostas Proedrou, Volodya Vovk, and Alex Gammerman. Inductive Confidence Machines for Regression. In Tapio Elomaa, Heikki Mannila, and Hannu Toivonen, editors, *Machine Learning: ECML 2002*, Lecture Notes in Computer Science, pages 345–356, Berlin, Heidelberg, 2002. Springer. ISBN 978-3-540-36755-0. [[↑ page 1](#)]
- [5] Jing Lei, Alessandro Rinaldo, and Larry Wasserman. A conformal prediction approach to explore functional data. *Annals of Mathematics and Artificial Intelligence*, 74(1):29–43, June 2015. ISSN 1573-7470. URL <https://doi.org/10.1007/s10472-013-9366-6>. [[↑ page 1](#)]
- [6] Ryan J. Tibshirani, Rina Foygel Barber, Emmanuel J. Candès, and Aaditya Ramdas. Conformal Prediction Under Covariate Shift, July 2020. URL <http://arxiv.org/abs/1904.06019>. [[↑ page 1](#)]
- [7] Rina Foygel Barber, Emmanuel J. Candès, Aaditya Ramdas, and Ryan J. Tibshirani. Conformal prediction beyond exchangeability, September 2022. URL <http://arxiv.org/abs/2202.13415>. [[↑ page 1](#)], [[↑ page 9](#)]
- [8] Adam Fisch, Tal Schuster, Tommi Jaakkola, and Regina Barzilay. Conformal Prediction Sets with Limited False Positives, February 2022. URL <http://arxiv.org/abs/2202.07650>. [[↑ page 1](#)]
- [9] Anastasios N. Angelopoulos, Stephen Bates, Adam Fisch, Lihua Lei, and Tal Schuster. Conformal Risk Control, April 2023. URL <http://arxiv.org/abs/2208.02814>. [[↑ page 1](#)]
- [10] Vladimir Vovk. Conditional validity of inductive conformal predictors, September 2012. URL <http://arxiv.org/abs/1209.2673>. [[↑ page 1](#)], [[↑ page 4](#)], [[↑ page 5](#)]
- [11] Yaniv Romano, Matteo Sesia, and Emmanuel Candès. Classification with Valid and Adaptive Coverage. In *Advances in Neural Information Processing Systems*, volume 33, pages 3581–3591. Curran Associates, Inc., 2020. URL <https://proceedings.neurips.cc/paper/2020/hash/244edd7e85dc81602b7615cd705545f5-Abstract.html>. [[↑ page 1](#)]
- [12] Jing Lei and Larry Wasserman. Distribution Free Prediction Bands, March 2012. URL <http://arxiv.org/abs/1203.5422>. [[↑ page 1](#)]
- [13] Jing Lei, Max G’Sell, Alessandro Rinaldo, Ryan J. Tibshirani, and Larry Wasserman. Distribution-Free Predictive Inference For Regression, March 2017. URL <http://arxiv.org/abs/1604.04173>. [[↑ page 1](#)], [[↑ page 2](#)], [[↑ page 4](#)], [[↑ page 6](#)]
- [14] Yaniv Romano, Evan Patterson, and Emmanuel J. Candès. Conformalized Quantile Regression, May 2019. URL <http://arxiv.org/abs/1905.03222>. [[↑ page 1](#)], [[↑ page 2](#)]

- [15] Leying Guan. Conformal prediction with localization, July 2020. URL <http://arxiv.org/abs/1908.08558>. [[↑ page 1](#)] [[↑ page 2](#)]
- [16] Leying Guan. Localized Conformal Prediction: A Generalized Inference Framework for Conformal Prediction, February 2022. URL <http://arxiv.org/abs/2106.08460>. [[↑ page 1](#)]
- [17] Xing Han, Ziyang Tang, Joydeep Ghosh, and Qiang Liu. Split Localized Conformal Prediction, February 2023. URL <http://arxiv.org/abs/2206.13092>. [[↑ page 1](#)] [[↑ page 2](#)] [[↑ page 3](#)] [[↑ page 6](#)]
- [18] Zhen Lin, Shubhendu Trivedi, and Jimeng Sun. Locally Valid and Discriminative Prediction Intervals for Deep Learning Models. In *Advances in Neural Information Processing Systems*, October 2021. URL [https://openreview.net/forum?id=xfDXFOI\\_bt](https://openreview.net/forum?id=xfDXFOI_bt). [[↑ page 1](#)] [[↑ page 2](#)] [[↑ page 6](#)]
- [19] Harris Papadopoulos, Alex Gammerman, and Volodya Vovk. Normalized nonconformity measures for regression Conformal Prediction. In *Proceedings of the 26th IASTED International Conference on Artificial Intelligence and Applications*, AIA '08, pages 64–69, USA, February 2008. ACTA Press. ISBN 978-0-88986-710-9. [[↑ page 1](#)] [[↑ page 2](#)]
- [20] Shai Feldman, Stephen Bates, and Yaniv Romano. Improving Conditional Coverage via Orthogonal Quantile Regression, October 2021. URL <http://arxiv.org/abs/2106.00394>. [[↑ page 1](#)] [[↑ page 3](#)] [[↑ page 4](#)]
- [21] Salim I. Amoukou and Nicolas J. B. Brunel. Adaptive Conformal Prediction by Reweighting Nonconformity Score, March 2023. URL <http://arxiv.org/abs/2303.12695>. [[↑ page 1](#)]
- [22] Rina Foygel Barber, Emmanuel J. Candes, Aaditya Ramdas, and Ryan J. Tibshirani. Predictive inference with the jackknife+, May 2020. URL <http://arxiv.org/abs/1905.02928>. [[↑ page 2](#)] [[↑ page 5](#)] [[↑ page 9](#)] [[↑ page S.8](#)]
- [23] Alethea Power, Yuri Burda, Harri Edwards, Igor Babuschkin, and Vedant Misra. Grokking: Generalization Beyond Overfitting on Small Algorithmic Datasets, January 2022. URL <http://arxiv.org/abs/2201.02177>. [[↑ page 5](#)]
- [24] PyNNDescent for fast Approximate Nearest Neighbors — pynndescent 0.5.0 documentation, 2020. URL <https://pynndescent.readthedocs.io/en/latest/>. [[↑ page 6](#)]
- [25] Wei Dong, Charikar Moses, and Kai Li. Efficient k-nearest neighbor graph construction for generic similarity measures. In *Proceedings of the 20th International Conference on World Wide Web*, pages 577–586, Hyderabad India, March 2011. ACM. ISBN 978-1-4503-0632-4. URL <https://dl.acm.org/doi/10.1145/1963405.1963487>. [[↑ page 6](#)]
- [26] Kexin Huang, Tianfan Fu, Wenhao Gao, Yue Zhao, Yusuf Roohani, Jure Leskovec, Connor Coley, Cao Xiao, Jimeng Sun, and Marinka Zitnik. Therapeutics Data Commons: Machine Learning Datasets and Tasks for Drug Discovery and Development. *Proceedings of the NeurIPS Track on Datasets and Benchmarks*, 1, December 2021. URL [https://datasets-benchmarks-proceedings.neurips.cc/paper\\_files/paper/2021/hash/4c56ff4ce4aaaf9573aa5dff913df997a-Abstract-round1.html](https://datasets-benchmarks-proceedings.neurips.cc/paper_files/paper/2021/hash/4c56ff4ce4aaaf9573aa5dff913df997a-Abstract-round1.html). [[↑ page 7](#)] [[↑ page S.2](#)]
- [27] David Weininger. SMILES, a chemical language and information system. 1. Introduction to methodology and encoding rules. *Journal of Chemical Information and Computer Sciences*, 28(1):31–36, February 1988. URL <https://doi.org/10.1021/ci00057a005>. [[↑ page 7](#)] [[↑ page S.2](#)]
- [28] Emily Engelhart, Ryan Emerson, Leslie Shing, Chelsea Lennartz, Daniel Guion, Mary Kelley, Charles Lin, Randolph Lopez, David Younger, and Matthew E. Walsh. A dataset comprised of binding interactions for 104,972 antibodies against a SARS-CoV-2 peptide. *Scientific Data*, 9(1):653, October 2022. ISSN 2052-4463. URL <https://www.nature.com/articles/s41597-022-01779-4>. [[↑ page 7](#)]

- [29] Yann LeCun and Corinna Cortes. The mnist database of handwritten digits. *undefined*, 2005. URL <https://www.semanticscholar.org/paper/The-mnist-database-of-handwritten-digits-LeCun-Cortes/dc52d1ede1b90bf9d296bc5b34c9310b7eaa99a2>. [<sup>↑</sup>page 8], [<sup>↑</sup>page S.3]
- [30] Michael Bian and Rina Foygel Barber. Training-conditional coverage for distribution-free predictive inference, January 2023. URL <http://arxiv.org/abs/2205.03647>. [<sup>↑</sup>page 9]
- [31] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011. [<sup>↑</sup>page S.2]
- [32] Seyone Chithrananda, Gabriel Grand, and Bharath Ramsundar. ChemBERTa: Large-Scale Self-Supervised Pretraining for Molecular Property Prediction, October 2020. URL <http://arxiv.org/abs/2010.09885>. [<sup>↑</sup>page S.2]
- [33] Murat Cihan Sorkun, Abhishek Khetan, and Süleyman Er. AqSolDB, a curated reference set of aqueous solubility and 2D descriptors for a diverse set of compounds. *Scientific Data*, 6(1):143, August 2019. ISSN 2052-4463. URL <https://www.nature.com/articles/s41597-019-0151-1>. [<sup>↑</sup>page S.3]
- [34] Anne Hersey. ChEMBL Deposited Data Set - AZ\_dataset. Technical report, EMBL-EBI, February 2015. URL <https://www.ebi.ac.uk/chembl/doc/inspect/CHEMBL3301361>. [<sup>↑</sup>page S.3]
- [35] Tobias H. Olsen, Iain H. Moal, and Charlotte M. Deane. AbLang: An antibody language model for completing antibody sequences. Preprint, Bioinformatics, January 2022. URL <http://biorxiv.org/lookup/doi/10.1101/2022.01.20.477061>. [<sup>↑</sup>page S.3]
- [36] Nicholas Carlini, Úlfar Erlingsson, and Nicolas Papernot. Distribution Density, Tails, and Outliers in Machine Learning: Metrics and Applications, October 2019. URL <http://arxiv.org/abs/1910.13427>. [<sup>↑</sup>page S.8]
- [37] Robert Baldock, Hartmut Maennel, and Behnam Neyshabur. Deep Learning Through the Lens of Example Difficulty. In *Advances in Neural Information Processing Systems*, volume 34, pages 10876–10889. Curran Associates, Inc., 2021. URL <https://proceedings.neurips.cc/paper/2021/hash/5a4b25aaed25c2ee1b74de72dc03c14e-Abstract.html>.
- [38] Vitaly Feldman. Does Learning Require Memorization? A Short Tale about a Long Tail, January 2021. URL <http://arxiv.org/abs/1906.05271>. [<sup>↑</sup>page S.8]
- [39] H. G. Landau. On dominance relations and the structure of animal societies: III The condition for a score structure. *The bulletin of mathematical biophysics*, 15(2):143–148, June 1953. ISSN 1522-9602. URL <https://doi.org/10.1007/BF02476378>. [<sup>↑</sup>page S.9]

---

# Adaptive Conformal Regression with Jackknife+ Rescaled Scores

---

## Supplementary Material

---

### S.1 Algorithms

In this section, we describe the calibration and PI prediction procedures for two versions of our method in the form of pseudocode. The simplest is algorithm S.1, where the kernel is assumed to be fixed. Indeed we've found in practice that using a KNN kernel with  $K = 10$  is an efficient and performant approach. We however also define algorithm S.2, where we tune  $N$  kernels on calibration data, with an objective motivated by the bound of theorem S.2. For high-dimensional data, such as when using embeddings to compute kernel similarities, we use a low-dimensional PCA and compute the sum of the marginal mutual information between the score and principal component. If a PCA is not used, this sum is an upper bound on the multi-dimensional mutual information, but we've also found that using 2-4 principal components captures an overwhelming majority of the data variance for our experiments when using latent space embeddings, and yields good empirical improvement when used for kernel tuning.

---

**Algorithm S.1** Jackknife+ rescaled score conformal regression without kernel tuning

---

**Input:**

- 1: Exchangeable data  $\{(X_i, y_i) \in \Omega_X \times \mathbb{R} \mid i \in [-T, \dots, N+1]\}$ .
  - 2: A learning algorithm  $\mathcal{A}$  mapping a sample of  $(X, y)$  pairs to a model  $m(X)$ .
  - 3: A kernel  $K : \Omega_X \times \Omega_X \rightarrow \mathbb{R}^+$ .
  - 4: A risk threshold  $\alpha \in [0, 1]$ .
  - 5:
  - 6: **procedure** PREDICTOR TRAINING
  - 7:      $m = \mathcal{A}\left(\{(X_i, y_i)\}_{i \in [-T, \dots, 0]}\right)$
  - 8: **end procedure**
  - 9: All free indices below  $(i, j, k)$  span  $[1, \dots, N]$ .
  - 10: **procedure** CALIBRATION
  - 11:     SET  $p_{ij} = K_{ij} / \sum_{k \neq i} K_{ik}$   $\triangleright K_{ij} = K(X_i, X_j)$
  - 12:     SET  $s_i^+ = |y_i - m(X_i)| / \sum_{k \neq i} p_{ik} s_k$
  - 13: **end procedure**
  - 14: **procedure** INTERVAL PREDICTION
  - 15:     SET  $Q = \lceil (1 - \alpha)(N + 1) \rceil$
  - 16:     SET  $p_{i(N+1)} = K_{i(N+1)} / \sum_{k \neq i, N+1} K_{ik}$
  - 17:     SET  $\hat{s}_{(N+1)i} = \sum_{k \neq i} p_{k(N+1)} s_k$
  - 18:     SORT  $S = [\hat{s}_{(N+1)1} s_1^+, \dots, \hat{s}_{(N+1)N} s_N^+]$
  - 19:     SET  $\Delta y_{N+1} = S[Q]$
  - 20: **end procedure**
- Output:** Test prediction interval  $[m(X) - \Delta y_{N+1}, m(X) + \Delta y_{N+1}]$ .
-

---

**Algorithm S.2** Jackknife+ with Kernel Tuning

---

```

1: procedure CALIBRATION WITH KERNEL TUNING( $n_{\text{PCA}}$ ,  $n_{\text{sample}}$ ,  $n_{\text{scan}}$ ,  $\beta_{\text{expand}} \geq 1$ )
2:   SAMPLE  $n_{\text{sample}}$  pairs of non-identical training inputs ( $X_a, X'_a$ )
3:   SET  $d_{\min}, d_{\max}$  to the extrema of  $\|X_a - X'_a\|$ 
4:   SET  $\lambda_n$  as  $n_{\text{scan}}$  values evenly space in logarithmic scale in  $[d_{\min}/\beta_{\text{expand}}, d_{\max} \times \beta_{\text{expand}}]$ 
5:   DEFINE  $N$  RBF kernels  $K_m$  with length scales  $l_m$ .
6:   for each  $\lambda_n$  do
7:     for  $m \in [1, \dots, N]$  do
8:       SET  $l_m = \lambda_n$ 
9:       SET  $p_{ij;m} = K_{ij;m} / \sum_{k \neq i, m} K_{ik;m}$   $\triangleright K_{ij;m} = K_m(X_i, X_j)$ 
10:      SET  $s_{i;m}^+ = |y_i - m(X_i)| / \sum_{k \neq i, m} p_{ik}s_k$ 
11:    end for
12:    SET  $\text{mi}_{mn} = \text{MI} \left( \{s_{i;m}^+\}_{i \neq m}, \{X_i\}_{i \neq m} \right)$ 
13:  end for
14:  SET  $n^*(m) = \underset{n}{\operatorname{argmin}} \text{mi}_{mn}$ 
15:  SET  $l_m = \lambda_{n^*(m)}$ 
16:  SET  $s_i^+ = s_{i;i}^+$ 
17: end procedure
18: procedure INTERVAL PREDICTION
19:   SET  $Q = \lceil (1 - \alpha)(N + 1) \rceil$ .
20:   SET  $p_{i(N+1)} = K_{i(N+1);i} / \sum_{k \neq i, N+1} K_{ik;i}$ 
21:   SET  $\hat{s}_{(N+1)i} = \sum_{k \neq i} p_{k(N+1)} s_k$ 
22:   SORT  $S = [\hat{s}_{(N+1)1} s_1^+, \dots, \hat{s}_{(N+1)N} s_N^+]$ 
23:   SET  $\Delta y_{N+1} = S[Q]$ 
24: end procedure
Output: Test prediction interval  $[m(X) - \Delta y_{N+1}, m(X) + \Delta y_{N+1}]$ .

```

---

## S.2 Dataset and Model Details

### S.2.1 One-Dimensional Regression

We formulate toy example of regression with label noise in one dimension as follows:

$$\begin{aligned} X &\sim U(0, 1), & y &= f(X) + \epsilon, & \epsilon &\sim \mathcal{N}(0, \lambda \nu(X)), \\ f(X) &= f_0 + X^2 \sin(k_f X + \phi_f), & \nu(X) &= \epsilon_0 + |\sin(k_\epsilon X + \phi_\epsilon)|, \end{aligned} \tag{8}$$

where

$$f_0 = 10^{-1}, k_f = 10, \phi_f = 1/2, \epsilon_0 = 10^{-2}, k_\epsilon = 2, \phi_\epsilon = 3 \times 10^{-1}, \lambda = 10^{-1}. \tag{9}$$

For each repetition of the experiment, we sample *i.i.d* pairs of  $(X, y)$  and split them into 1000 training points, 10000 test points and a variable number of calibration points. A random-forest regression model is trained, using the default parameters of SCIKIT-LEARN v1.2.2 [31] for our base model, or disabling bootstratpping to produce an overfitting model. For all methods, kernel scores are evaluated on the raw input values  $X$

### S.2.2 Chemical Property Regression on TDC Datasets

We use two datasets from the Therapeutics Data Commons repository [26], each of which is treated independently. Both datasets contain samples consisting of pairs of SMILES [27], descriptions of chemical structure as text sequences, and a target value for the property of interest. The datasets are already divided into training, validation and test samples, however, we merge the validation and test data and re-divide it randomly into calibration and test to allow for enough statistics. For each task, we fine tune a CHEMBERTA [32] language model pretrained on SMILES language modelling, provided by HUGGINGFACE.

The first task is to predict drug solubility on data originally produced by Sorkun et al. [33] and contains 9982 samples, split into 6988 training points and 2994 test points.

The second task consists of estimating the drug microsome clearance (drug elimination rate by the liver) on 1102 samples from Hersey [34], which are split into 772 training points and 330 test points.

Our fine-tuned CHEMBERTA models are defined HUGGINGFACE AutoModelForSequenceClassification<sup>5</sup> with DeepChem/ChemBERTa-77M-MTR weights and sequence data is processed with the adapted AutoTokenizer, with maximum sequence lengths set to the maximum sequence length in the training data. The training is performed with the mean-squared-error loss, using a batch size of 64 and a learning rate of  $4.0 \times 10^{-5}$  over 100 epochs.

Embeddings for kernel scores are computed using the output of the `classifier.dense` layer of the model, which is the first linear layer of the classification head of the fine-tuned model.

### S.2.3 AlphaSeq Antibody Affinity Regression

We use the data from the AlphaSeq survey of 104,972 measurements of the binding affinity of antibody proteins to a SARS-CoV-2 target peptide, paired with amino-acid sequence information for the antibodies. We pre-process the label data by dropping missing measurements and measurements on reference epitopes (keeping only those labelled as MIT\_Target), resulting in 69,297 valid sequence-affinity pairs. The data is divided into a training/validation/test+calibration split of sizes 39466/12517/17314. The test+calibration set is randomly subsampled into a test dataset of size 7314 and a calibration dataset of variable size.

We use the amino-acid sequences for both the heavy and the light chains of the antibody as inputs to pre-trained, frozen-weight ABLANG [35] amino-acid language models adapted to each chain type. The embeddings thus produced are concatenated and mapped to numerical predictions with a two-hidden-layer neural network with layer widths (128, 32) and ReLU activations. The final single-output layer does not have an activation function. This model is trained with the Adam optimizer using a learning rate of  $1.0 \times 10^{-5}$ , a batch size of 128 and is regularized with early stopping, monitoring the validation mean-squared error.

Embeddings for kernel scores are the concatenated outputs of the ABLANG models.

### S.2.4 MNIST Regression

We use the classic MNIST dataset [29] repurposed as a regression task where each digit is labelled by its floating point value. We use data augmentation both at training time and at test time to increase the potential confusion between similar digits, which leads to a non-trivial error structure as we discuss in appendix S.3.4.

We use the following randomized distortions, applied sequentially

- Gaussian blurring with kernel size  $3 \times 3$ , applied with probability 30%.
- Perspective transformation with scale 0.4, applied with probability 30% (`torchvision RandomPerspective`).
- Gaussian noise on each pixel with mean 0 and standard deviation  $1/(6 + \nu)$  where  $\nu \sim U(0, 5)$ .

The specific choice of transformation is quite arbitrary, but is meant to ensure qualitatively that numbers are nearly always recognizable to human observers, while making significant distortions common. Transformations are resampled every time an image is used.

We train a convolutional neural network (CNN) on this regression task by optimizing the mean squared error loss with an ADAM optimizer with learning rate  $5 \times 10^{-4}$ . The CNN has the architecture described in fig. S.1.

Embeddings for kernel scores are the 256-dimensional outputs of the hidden linear layer.

---

<sup>5</sup>These models can also do regression, despite the name

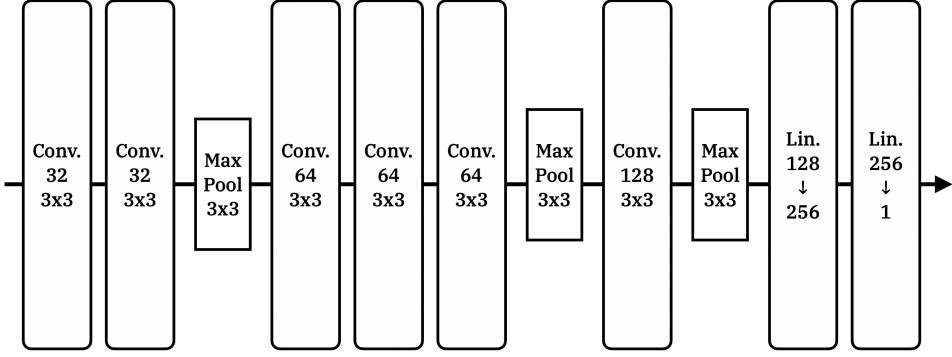


Figure S.1: CNN used to perform the MNIST regression task. Each hidden layer is followed by a batch normalisation layer and a rectified linear unit activation function.

### S.3 Additional Experimental Results

#### S.3.1 Original Kernel Tuning Strategies

We present in table 2 an extended version of table 1 including both our re-tuned kernels for the baselines, as described in section 4.1 and the original tuning techniques proposed by each method. We report uncertainties as the  $1 - \sigma$  percentile error, *i.e.* the 67<sup>th</sup> percentile of absolute deviation from the mean, evaluated with bootstrapping. This is equivalent to reporting the standard deviation if the metrics are normally distributed, which is a weakly-motivated hypothesis given our limited statistics. As we describe in the body of the manuscript, the original tuning strategy proposed for LVD leads to many infinite intervals, sometimes a large majority. Given that it does reach reasonable performance in the 1D example, and that the worse case is ALPHASEQ, which uses very high-dimensional embeddings, we conjecture that this reflects the unsuitability of this tuning procedure in high dimensions.

#### S.3.2 Jackknife+ Kernel Tuning

For the sake of limiting computational costs and brevity, we demonstrated the performance of our method with a fixed 10-NN kernel. In this section, we compare this approach with a RBF approach, where the length scale is tuned using algorithm S.2, limiting ourselves to the one-dimensional example.

As we show in table 3, a tuned RBF kernel yields comparable results to the KNN approach in the low statistics regime, but achieves better results when more data is available. As we use a Kozachenko-Leonenko-based mutual information estimator, the improved performance in the higher statistics regime is likely due to its asymptotically vanishing bias.

This potential for squeezing extra performance is confirmed by investigating the tuning procedure. In fig. S.2, we show that there is a finite range where  $\text{MI}(s^+, X)$  is minimized, and that this minimizer does improve over the same measure evaluated on the 10-NN kernel.

#### S.3.3 AlphaSeq Prediction Details

As we discuss in section 4.2.1, the performance of the baselines is rather poor on the AlphaSeq dataset, which might be especially surprising for LVD due to the large calibration sets. The difficulties of MADSPLIT can be attributed to the clearly observable overfitting of our trained model, which shows in the form of an S-shape of the label-prediction plot of the test data, as seen in fig. S.3 (left). We attribute the degraded performance of LVD to the structure of data in the latent space used for similarity measurements: as we show in fig. S.3 (right), a UMAP of the test data highlights that despite much of the data being concentrated in large clusters, there is a significant number of isolated communities, which are particularly troublesome for LVD. Of course, UMAP representations can obfuscate many features of the data layout, and this explanation is only tentative.

Dataset	$N_{\text{cal}}$	Method	Cov. ( $\gtrsim 0.95$ )	$\text{IS} \downarrow$	$R^2_{SQI} \uparrow$	$\tau(SQI) \uparrow$	$\tau(SI) \uparrow$
1D	500	Ours	0.96(1)	0.18(1)	0.87(3)	0.91(4)	0.35(1)
		MADSPLIT (KNN)	0.952(9)	0.18(1)	0.80(4)	0.96(3)	0.344(9)
		MADSPLIT (Median)	0.95(1)	0.19(1)	$\ll 0$	0.1(1)	0.04(1)
		LVD (Tuned)	0.982(3)	0.21(1)	0.61(9)	0.86(4)	0.33(1)
		LVD (Base)	0.985(2)	0.22(1)	0.5(1)	0.79(9)	0.31(2)
	2000	Ours	0.952(2)	0.171(5)	0.90(2)	0.96(4)	0.36(1)
		MADSPLIT	0.951(3)	0.184(2)	0.81(2)	0.93(6)	0.340(7)
		MADSPLIT (Median)	0.95(1)	0.19(1)	$\ll 0$	0.1(1)	0.04(1)
		LVD (Tuned)	0.968(3)	0.174(3)	0.92(2)	0.96(3)	0.361(6)
		LVD (Base)	0.985(2)	0.22(1)	0.5(1)	0.79(9)	0.31(2)
TDC Sol.	400	Ours	0.95(1)	5.3(5)	0.4(1)	0.6(1)	0.09(1)
		MADSPLIT (KNN)	0.95(1)	5.0(3)	0.53(5)	0.71(6)	0.182(3)
		MADSPLIT (Median)	0.95(1)	4.6(3)	$\ll 0$	0.49(2)	0.127(4)
		LVD (Tuned)	0.958(7)	5.2(3)	0.3(5)	0.7(1)	0.07(5)
		LVD (Base)	0.989(2)	9(1)	$-0.8(2)$	0.6(1)	0.06(1)
	800	Ours	0.96(1)	5.2(4)	0.5(1)	0.7(1)	0.11(1)
		MADSPLIT (KNN)	0.951(6)	5.1(2)	0.58(7)	0.71(5)	0.179(6)
		MADSPLIT (Median)	0.95(1)	4.7(2)	$\ll 0$	0.51(7)	0.124(8)
		LVD (Tuned)	0.960(9)	5.4(2)	0.4(3)	0.77(7)	0.07(1)
		LVD (Base)	0.978(5)	7.9(5)	$-0.1(1)$	0.8(1)	0.058(7)
TDC Clear.	60	Ours	0.95(2)	$1.9(3) \cdot 10^2$	0.2(3)	0.6(1)	0.37(3)
		MADSPLIT (KNN)	0.95(2)	$1.9(5) \cdot 10^2$	0.2(3)	0.60(9)	0.39(1)
		MADSPLIT (Median)	0.96(2)	$2.0(1) \cdot 10^2$	$\ll 0$	0.5(1)	0.39(1)
		LVD (Tuned)	0.97(2)	$2.2(2) \cdot 10^2$	$\ll 0$	$-0.3(6)$	0.1(2)
		LVD (Base)	0.98(1)	$2.3(2) \cdot 10^2$	$\ll 0$	$-0.0(3)$	0.00(5)
	240	Ours	0.96(1)	$1.65(9) \cdot 10^2$	0.3(2)	0.5(1)	0.39(7)
		MADSPLIT (KNN)	0.96(1)	$1.64(8) \cdot 10^2$	0.41(5)	0.6(1)	0.41(7)
		MADSPLIT (Median)	0.96(1)	$1.94(2) \cdot 10^2$	$\ll 0$	0.50(7)	0.41(5)
		LVD (Tuned)	0.97(1)	$2.02(5) \cdot 10^2$	$\ll 0$	0.1(2)	0.1(1)
		LVD (Base)	0.995(6)	$2.57(3) \cdot 10^2$	$\ll 0$	$-0.1(1)$	$-0.03(4)$
$\alpha$ Seq CoVID	2000	Ours	0.952(6)	4.5(1)	0.24(8)	0.5(1)	0.05(1)
		MADSPLIT (KNN)	0.952(5)	4.6(1)	0.1(2)	0.2(2)	0.01(2)
		MADSPLIT (Median)	0.950(5)	4.06(7)	$\ll 0$	0.2(3)	0.04(4)
		LVD (Tuned)	0.994(1)	7.6(5)	$\ll 0$	0.1(1)	$-0.004(6)$
		LVD (Base)	0.9997(3)	8.6(8)	$\ll 0$	0.0(2)	$-0.003(6)$
	10000	Ours	0.953(4)	4.54(6)	0.28(6)	0.6(1)	0.052(8)
		MADSPLIT (KNN)	0.952(2)	4.59(6)	0.0(1)	0.2(2)	0.01(1)
		MADSPLIT (Median)	0.951(5)	4.05(2)	$\ll 0$	0.1(4)	0.01(7)
		LVD (Tuned)	0.973(4)	5.49(4)	$-0.26(4)$	0.5(1)	0.052(8)
		LVD (Base)	0.99993(6)	9.54(3)	$\ll 0$	0.0(3)	$-0.001(9)$

Table 2: Conformal interval prediction performance metrics on N benchmark datasets. Numbers in parentheses are the uncertainty on the last significant digit evaluated over 10 random samples of test and calibration data.

Dataset	$N_{\text{cal}}$	Method	Cov. ( $\gtrsim 0.95$ )	$\text{IS} \downarrow$	$R^2_{SQI} \uparrow$	$\tau(SQI) \uparrow$	$\tau(SI) \uparrow$
1D	500	Ours (KNN)	0.95(1)	0.173(9)	0.85(4)	0.89(7)	0.35(1)
		Ours (tuned)	0.95(1)	0.17(1)	0.84(8)	0.85(6)	0.34(2)
	2000	Ours (KNN)	0.951(5)	0.174(6)	0.88(2)	0.91(4)	0.362(8)
		Ours (tuned)	0.949(3)	0.157(4)	0.93(1)	0.92(4)	0.37(1)

Table 3: PI metrics evaluated on the 1-dimensional regression task described in section 4.2, comparing our approach with a fixed 10-NN kernel and a RBF kernel tuned with algorithm S.2.

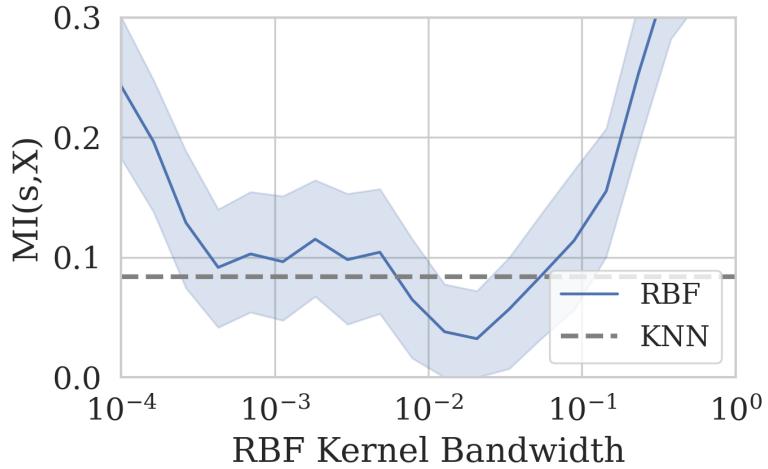


Figure S.2: Dependence of the mutual information between  $s^+(X, y)$  and  $X$  as a function of the kernel length scale, reported as the average over each Jackknife+ split.

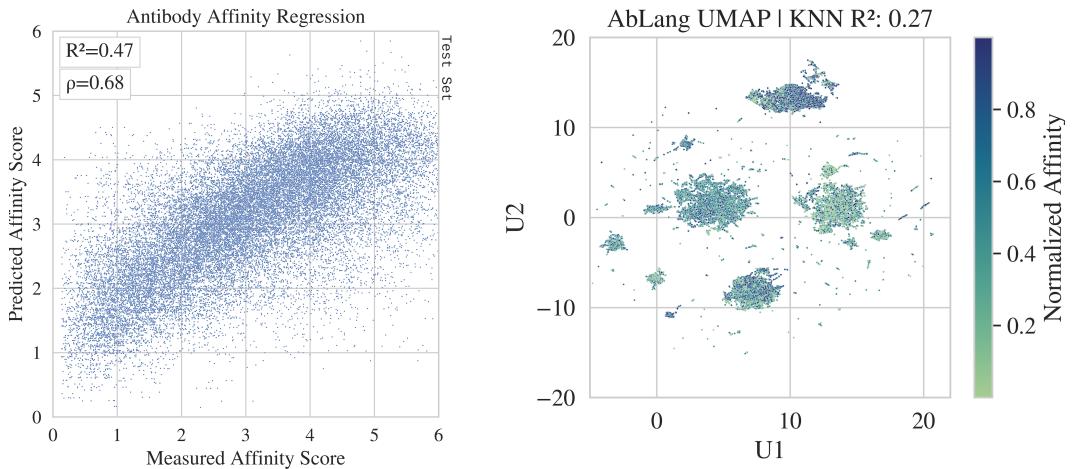


Figure S.3: Visualization of our conjectured explanations for the difficulty of MADSPLIT and LVD on the ALPHASEQ dataset. (Left) Regression evaluation plot showing the systematic over and under prediction at the lower and higher edges of the label distribution. (Right) UMAP visualization of the ABLANG embeddings used for kernel similarity scores illustrating the scattered structure of the dataset.

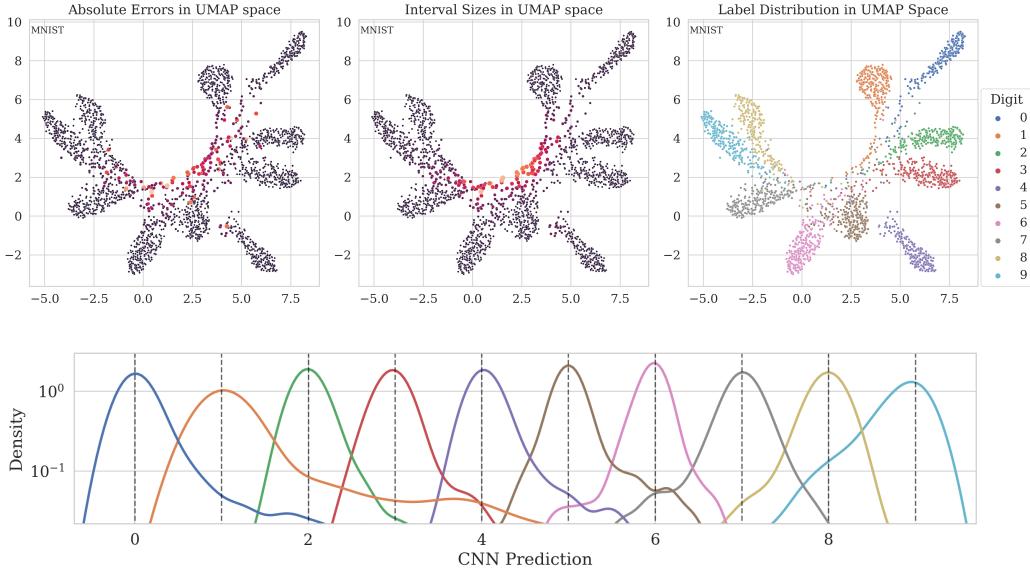


Figure S.4: (Top) Latent-space structure for the MNIST Regression task interval sizes and labels visualized with UMAP. (Bottom) Prediction distribution per true label.

### S.3.4 Analysis of the MNIST Regression Task Error Distribution

The MNIST Regression task stands out as the only one where no approach yields even a positive  $R_{SQI}^2$ , despite achieving rather high correlation scores (table 1).

This observation can be explained by the reason that motivated our choice of this dataset, despite its seemingly contrived nature: the bulk and the tail of the per-label error distributions can be expected to be governed by two independent phenomena. On the one hand, most images are associated with a floating-point label that is distributed around the correct value<sup>6</sup>, as can be seen in fig. S.4 (bottom), where the predictions for each digit have similar distributions close to the correct value. On the other hand, this same figure shows that the tail of the error distribution is very class-dependent. This tail structure is dictated by the confusion between digits induced by our test-time data augmentation, which can be observed in latent space, as shown in fig. S.4 (top): most of each class is well separated from the others, but each class has a tail overlapping with other classes, inducing high errors. Different digits have different fractions of confusing examples as well as different possible error classes, leading to the observed error dependence. As a result, our hypothesis that the conditional error quantiles is essentially some input-dependent rescaling of the conditional mean is badly broken. The error mean nevertheless captures some dependence of the error on the inputs, but the absolute interval size is thrown off, which is reflected in the high correlation, but low  $R_{SQI}^2$ .

## S.4 The Mean Interval Size is not a Sufficient Measure of Adaptivity

We argue that average interval sizes (IS) are a crude metric for measuring the adaptivity of a PI prediction method. Indeed, like any average-based metric, IS is sensitive to large outliers: a minority population with large errors would have little impact on the PI of a non adaptive method while a dynamic PI prediction would assign larger interval to this sub-population and lead to an increase in IS.

To illustrate this point Let us consider a regression dataset with  $N$  elements whose regression errors follow a half normal distribution  $|\mathcal{N}(0, \sigma)|$ , where a fraction  $\beta$  of points have scale factor  $\sigma = \sigma_0$ , while the other  $(1 - \beta)$  have  $\sigma = \lambda\sigma_0$ . If  $N, \lambda \gg 1$  and  $\beta > 0.95$ , the standard conformal

<sup>6</sup>This spread is essentially dictated by the equilibrium reached between minimizing the MSE on clearly-identifiable examples, model capacity, and regularisation.

regression intervals based on absolute errors will be dictated by the normal errors and will therefore yield interval sizes close to twice the 95<sup>th</sup> of the half-normal distribution:

$$IS_{\text{flat}} = 2 \times 1.96\sigma_0 . \quad (10)$$

On the other hand, a perfectly adaptive PI predictor would yield this same  $2 \times 1.96\sigma_0$  interval size for the  $\sigma = \sigma_0$  population and  $2 \times 1.96\lambda\sigma_0$  for the others. The average interval size of the "perfect" adaptive PI would therefor be

$$IS_{\text{perfect}} = 2 \times 1.96\sigma_0 (\beta + (1 - \beta)\lambda) , \quad (11)$$

which will be larger than the flat average interval sizes for large enough  $\lambda$ .

We illustrate this point empirically in fig. S.5. As expected, the optimally adaptive PI increases when non-coverage risk  $\alpha$  is comparable with the fraction of high error examples. This means that IS is inappropriate to use when the error distribution has long tail due to rare difficult examples, which is a common situation in image processing for example [36–38], or due to skewed heteroscedastic errors.

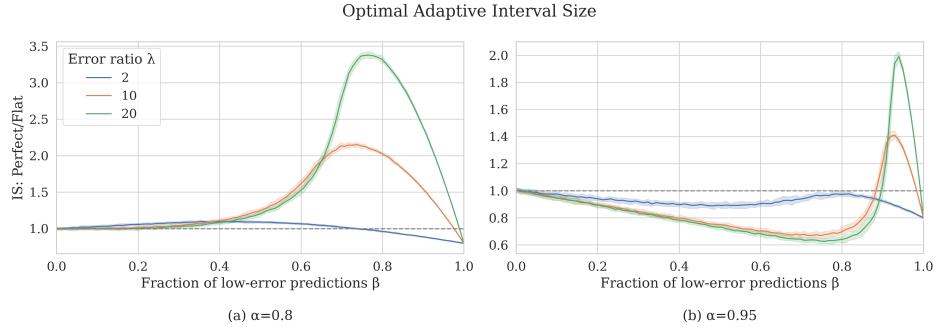


Figure S.5: Average interval size ratio between a flat conformal and a perfectly adaptive PI when errors are normally distributed, with  $\sigma = \sigma_0$  for a fraction  $\beta$ , and  $\sigma = \lambda\sigma_0$  for the remaining  $1 - \beta$ .

It is usually desirable to reduce IS at constant coverage, but this reduction is not necessarily a signal of improved adaptivity: the PI predictor might be systematically miscovering high-error minority population, favoring low-error minorities. This observation is what motivates our proposal for more correlation-oriented metrics in section 2.2, which are sensitive to whether coverage distributes uniformly across high and low error populations.

## S.5 Formal Global Coverage Guarantees

Here we show how the conformal intervals defined in eq. (6) provides the following theoretical guarantee:

$$P(Y_{N+1} \in C^{+\alpha}(X_{N+1})) \geq 1 - 2\alpha. \quad (12)$$

The proof is nearly identical to that provided in Section 6 of Barber et al. [22] and follows in four parts:

- We define a matrix of score competitions between all the  $\{s_{ij}^+\}_{i,j \in [1..N+1]}$ , whose rows and columns are distributionally permutation-invariant.
- We use a theorem from Landau to show that there is an upper bound on the number of points that win atypically many competitions
- We use the distributional permutation invariance of the competition matrix to show that there is an upper bound on the probability that the test point  $Z_{N+1}$  is such an “atypical winner”.
- We show that  $Y_{N+1} \notin C^{+\alpha}$  implies that  $Z_{N+1}$  is an atypical winner, therefore obtaining an upper bound on the probability of this event by contraposition.

Let's get through each one

### S.5.1 Score and competition matrices

Let us define the following score matrix that has manifest distributional permutation invariance due to the exchangeability of  $X_1, \dots, X_{N+1}$ :

$$R = (R_{ij}) = \begin{cases} s_{ij}^+ \text{ for } i \neq j \in [1..N+1] \\ R_{ii} = \infty \end{cases} \quad (13)$$

From this matrix, be further build a competition matrix

$$A_{ij} = \text{Indicator}(R_{ij} > R_{ji}). \quad (14)$$

Following the original proof, we define the set of “strange” points  $S(A)$  as those that win abnormally many competitions:

$$S(A) = \left\{ i \in [1..N+1] \mid \sum_j A_{ij} \geq (1 - \alpha)(N + 1) \right\}. \quad (15)$$

### S.5.2 Bounding strange points

There is a finite budget of victories in  $A$  since  $A_{ij} = 1 \Leftrightarrow A_{ji} = 0$ , so there is an upper bound on the size of  $S(A)$ . This is formalized in a theorem from Landau [39] that implies

$$|S(A)| \leq 2\alpha(N + 1). \quad (16)$$

### S.5.3 From set sizes to probabilities

The matrix  $R$ , and therefore  $A$  is distributionally permutation invariant, meaning that for any permutation matrix  $\Pi$  and any possible matrix value  $A_0$ ,  $P(A = A_0) = P(A = \Pi A_0 \Pi^T)$ . Permutations on the rows of  $A$  correspond to permuting the  $X_i$  so that

$$P(X_{N+1} \in S(A)) = P(X_j \in S(\Pi_{j,N+1} A \Pi_{j,N+1}^T)) = P(X_j \in S(A)), \quad (17)$$

where  $\Pi_{j,N+1}$  is a permutation matrix exchanging rows  $j$  and  $N + 1$ . Therefore the probability that  $N + 1 \in S(A)$  is

$$\frac{\langle |S(A)| \rangle}{N + 1} \leq 2\alpha, \quad (18)$$

where the inequality follows from the bound on  $|S(A)|$ .

#### S.5.3.1 Connecting strange points to coverage

Let us suppose that  $X_{N+1}$  is not covered by its interval  $C^{+\alpha}$ , this implies that

$$|\mu(X_{N+1}) - Y_{N+1}| > q^\alpha \left( \left\{ \sigma_{i,N+1} \frac{|\mu(X_i) - Y_i|}{\sigma_{N+1,i}} \right\} \right), \quad (19)$$

**i.e.** that for at least  $(1 - \alpha)(N + 1)$  indices  $j \in [1..N]$ , we have  $s_{(N+1)j}^+ \geq s_{j(N+1)}^+$ , thus making  $X_{N+1}$  a strange point, which has probability bounded by  $1 - 2\alpha$ , therefore proving the coverage guarantee we announced.

## S.6 Method-Agnostic Results on Local Coverage

### S.6.1 Position-independent score distributions guarantee strong input-space local coverage

The intuition behind our localized scores  $s^+(X, y)$  is that we try to build scores that are not sensitive to local variations of the size of errors so that we can use the whole calibration dataset while getting prediction intervals that are tuned to the local error scale.

This can be formalized as follows: our goal with  $s^+$  is to ensure that given a random variable  $(X, y) \sim \pi(X, y)$ , we have  $X \perp s^+(X, y)$ . This is actually a sufficient condition for strong input-space local coverage:

#### Proposition S.1

Consider the data  $M_{X \times y}, \mu, s, S^{(\alpha)}$  defined in section 1.3.

Let  $(X, y) \sim \pi$  be a random variable, then

$$X \perp s(X, y) \Rightarrow \alpha - \text{ISCC}.$$

*Proof of proposition S.1.* Let us remember how the conformal sets  $S^{(\alpha)}$  are obtained: we sample a size  $n$  calibration set  $X_i, y_i \stackrel{\text{i.i.d.}}{\sim} \pi$ , define  $\hat{q}_n^{(\alpha)}$  as the  $\lceil (1 - \alpha)(n + 1) \rceil / (n + 1)$ -th empirical quantile of the empirical distribution of  $\{s(X_i, y_i)\}_{1 \leq i \leq n}$  and define  $S^\alpha(X) = \{\hat{y} \in \Omega_y | s(X, \hat{y}) \leq \hat{q}_n^{(\alpha)}\}$ .

Having  $(X, y) \perp (X_i, y_i)$  and  $(X, y) \sim \pi$  ensures the global coverage probability:

$$\mathbb{P}(y \in S^{(\alpha)}(X)) \geq 1 - \alpha. \quad (20)$$

Let us assume  $X \perp s(X, y)$ . Under our independence assumption, the conditional PDF of  $s(X, y)$  verifies  $P(s(X, y) | X) = P(s(X, y))$ , or

$$\begin{aligned} \forall \omega_X \in \mathcal{F}_X, \omega_y \in \mathcal{F}_y, \sigma \in \mathcal{B}(\mathbb{R}), \\ \mathbb{P}(s(X, y) \in \sigma_{Xy}) = \mathbb{P}(s(X, y) \in \sigma | X) \end{aligned} \quad (21)$$

so that  $\mathbb{P}(s(X, y) \leq q_n^{(\alpha)} | X \in \omega_X) = \mathbb{P}(s(X, y) \leq q_n^{(\alpha)})$ .

Given that  $y \in S^{(\alpha)} \Leftrightarrow s(X, y) \leq q_n^{(\alpha)}$  and  $\mathbb{P}(y \in S^{(\alpha)}(X)) \geq 1 - \alpha$ , we find the desired property holds under our assumptions:

$$\forall \omega_X \in \mathcal{F}_X, \mathbb{P}_{X, y \sim \pi}(y \in S_{\pi, \mu}^{(\alpha)}(X) | X \in \omega_X) \geq 1 - \alpha. \quad (22)$$

□

### S.6.2 Extending local coverage guarantees to imperfect independence

In practice, the independence of score and inputs will never be realized perfectly. We can nevertheless provide a bound on the local coverage based on their degree of independence, measured as a statistical distance between their joint and product distributions  $p_{Xs}(X, s(X, y))$  and  $p_X(X) \otimes p_s(s(X, y))$ .

#### Theorem S.2

Consider the data  $M_{X \times y}, \mu, s, S^{(\alpha)}$  defined in section 1.3.

Let  $(X, y) \sim \pi$  be a random variable. Let  $\text{MI}_{Xs} = \text{MI}(X, s(X, y))$  be the mutual information and assume  $0 < \text{MI}_{Xs} < \infty$ , then on any  $\omega_X \in \mathcal{F}_X$  such that  $0 < \mathbb{P}(X \in \omega_X) < \infty$ ,

$$\mathbb{P}(y \in S(X) | X \in \omega_X) \geq (1 - \alpha) - \frac{\sqrt{1 - \exp(-\text{MI}_{Xs})}}{\mathbb{P}(X \in \Omega(X))}. \quad (23)$$

Note that this implies the simpler

$$\mathbb{P}(y \in S(X) | X \in \omega_X) \geq (1 - \alpha) - \frac{\sqrt{\text{MI}_{Xs}}}{\mathbb{P}(X \in \Omega(X))}, \quad (24)$$

which is a pretty tight approximation for small  $\text{MI}_{Xs}$  (better than 4% for  $\text{MI}_{Xs} \leq 0.3$ ) but becomes vacuous faster for large values.

This theorem follows from the Bretagnolle-Huber theorem (see below) and the following lemma:

**Lemma S.3** (main technical result)

Let  $p_{Xs}$  be the probability density of  $(X, s(X, y))$  and  $p_X, p_s$  the marginal densities of  $X$  and  $s(X, y)$ . If  $p_{Xs}(X, s)$  and  $p_X(X) \otimes p_s(s)$  have finite total variation  $\delta_{Xs}$ , then on any  $\omega_X \in \mathcal{F}_X$ ,

$$|\mathbb{P}(y \in S^\alpha(X) | X \in \omega_X) - (1 - \tilde{\alpha})| \times \mathbb{P}(X \in \omega_X) \leq \delta_{Xs} \quad (25)$$

where

$$1 - \tilde{\alpha} = \frac{\lceil (1 - \alpha)(n + 1) \rceil}{n + 1}. \quad (26)$$

**Bretagnolle-Huber Theorem**

Given two probability distributions  $P$  and  $Q$  such that  $P \ll Q$ , then their total variation  $\delta(P, Q)$  verifies

$$\delta(P, Q) \leq \sqrt{1 - \exp(-D_{KL}(P||Q))}. \quad (27)$$

In particular, given two random variables  $X, Y$ , we have a bound on the total variation between their joint and product distributions expressed in terms of their mutual entropy:

$$\delta(p(X, Y), p(X)p(Y)) \leq \sqrt{1 - \exp(-\text{MI}(X, Y))}. \quad (28)$$

Let us now prove lemma S.3, which is the real meat of the result.

*Proof of lemma S.3.* Let  $\omega_X \in \mathcal{F}_X$  such that  $0 < |\omega_X| < \infty$ . Let furthermore  $q \in [0, 1]$  represent the quantile value used to achieve  $(1 - \alpha)$  global coverage. We define  $\tilde{\alpha}$  such that  $\mathbb{P}(s(X, y) \leq q) = 1 - \tilde{\alpha}$ , i.e.

$$(1 - \tilde{\alpha}) = \frac{\lceil (1 - \alpha)(n + 1) \rceil}{n} \geq (1 - \alpha). \quad (29)$$

Let us consider the following quantity, writing  $s$  as shorthand for  $s(X, y)$ :

$$\Delta(\omega_X) = \left| \int_{\omega_X} dX \int_0^q ds p_{Xs}(X, s) - p_X(X)p_s(s) \right| \quad (30)$$

the integrand is the absolute difference of probabilities over the measurable set  $\omega_X \times [0, q]$ . By definition it is smaller than its supremum over all measurable sets; therefore

$$\Delta(\omega_X) \leq \delta_{Xs}. \quad (31)$$

Furthermore,

$$\Delta(\omega_X) = \left| \mathbb{P}(s \leq q, X \in \omega_X) - (1 - \tilde{\alpha})\mathbb{P}(X \in \omega_X) \right| \quad (32)$$

$$= \left| \mathbb{P}(s \leq q | X \in \omega_X) - (1 - \tilde{\alpha}) \right| \mathbb{P}(X \in \omega_X). \quad (33)$$

□

The bound becomes vacuous on very unlikely sets, which seems unavoidable. This limitation is related to the fact that  $\delta$  or  $\text{MI}$  are global measures and that the contribution from any small set is small. Therefore, a small set can deviate from the mean by a factor that is inversely proportional to its probability. Nevertheless, decreasing the mutual information uniformly decreases the penalty in the local coverage. We have furthermore observed empirically that minimizing the score-input mutual information four our method improves the PI evaluation metrics, even when the bound derived from the minimum is very weak.

## S.7 Rescaled Scores and Concentration Inequalities

In this section, we provide "inspirational" inequalities that result from straightforward applications of concentration inequalities to mean-rescaled scores. These inequalities cannot be used to put bounds on the local coverage properties of CR methods, but they give hints of why rescaled scores improve local coverage, and of possible directions to explore to obtain valid local bounds.

Throughout this section, we take a MADSPLIT-like approach in the sense that we assume that we can directly put a threshold on  $s(X)/\hat{s}(X)$  to obtain PI. Note that we nevertheless assume perfect moment estimators, while finite-sample kernel methods have non-zero bias and variance, so that the inequalities here do not apply for finite samples.

**Markov Local Coverage Guarantee** This is probably the most straightforward inequality: assume that we have obtained a threshold  $q^\alpha$  by any method. Markov's concentration inequality ( $\mathbb{P}(Z > a) \leq \bar{Z}/a$  for  $Z > 0$ ) applied to the random variable  $s(X, y)$  with fixed  $X$  implies

$$\mathbb{P}\left(\frac{s(X, y)}{\mathbb{E}_y s(X, y)} \leq q^\alpha | X\right) \geq 1 - \frac{1}{q^\alpha}, \quad (34)$$

which is an actual local coverage guarantee (assuming a perfect conditional mean estimator). Provided it can be extended to an actual mean estimator, this bound might be competitive for error distributions with long tails.

**Cantelli's Inequality** Let us consider Cantelli's inequality:

$$\mathbb{P}(X - \bar{X} \geq \lambda) \leq \frac{\sigma_X^2}{\sigma_X^2 + \lambda^2}. \quad (35)$$

Applying this to the random variable  $s(X, y)$  with fixed  $X$  and  $\lambda = (1 - \tau)\bar{s}(X)$ , it can be rearranged as

$$\mathbb{P}\left(\frac{s(X, y)}{\bar{s}(X)} \geq \tau\right) \leq \frac{\sigma_s^2(X)}{\sigma_s^2(X) + (\tau - 1)^2 \bar{s}(X)}, \quad (36)$$

where  $\bar{s}(X)$  and  $\sigma_s^2(X)$  are the  $X$ -conditional mean and variance of  $s$ .

Solving for  $\tau$  so that the right-hand-side equals  $\alpha$ , we find the following inequality

$$\mathbb{P}\left(\frac{s}{\bar{s}(X)} \geq 1 + \frac{\sigma_s(X)}{\bar{s}(X)} \sqrt{\frac{1 - \alpha}{\alpha}}\right) \leq \alpha. \quad (37)$$

This inequality shows that we can reformulate the hypotheses of our method on the  $X$ -independence of the ratio  $q^{1-\alpha}(s|X)/\bar{s}(X)$  for others on  $\sigma_s(X)/\bar{s}(X)$ . If the latter is  $X$ -independent, the inequality above shows the existence of a threshold on  $s/\bar{s}(X)$  that guarantees local coverage.