

---

# Few-shot Conformal Prediction with Auxiliary Tasks

---

Adam Fisch<sup>1</sup> Tal Schuster<sup>1</sup> Tommi Jaakkola<sup>1</sup> Regina Barzilay<sup>1</sup>

## Abstract

We develop a novel approach to conformal prediction when the target task has limited data available for training. Conformal prediction identifies a small set of promising output candidates in place of a single prediction, with guarantees that the set contains the correct answer with high probability. When training data is limited, however, the predicted set can easily become unacceptably large. In this work, we obtain substantially tighter prediction sets while maintaining desirable marginal guarantees by casting conformal prediction as a meta-learning paradigm over exchangeable collections of auxiliary tasks. Our conformalization algorithm is simple, fast, and agnostic to the choice of underlying model, learning algorithm, or dataset. We demonstrate the effectiveness of this approach across a number of few-shot classification and regression tasks in natural language processing, computer vision, and computational chemistry for drug discovery.

## 1 Introduction

Accurate estimates of uncertainty are important for difficult or sensitive prediction problems that have variable accuracy (Amodei et al., 2016; Jiang et al., 2012; 2018; Angelopoulos et al., 2021). Few-shot learning problems, in which training data for the target task is severely limited, pose a discouragingly compounded challenge: in general, not only is (1) making accurate predictions with little data hard, but also (2) rigorously quantifying the uncertainty in these few-shot predictions is even harder.

In this paper, we are interested in creating confident *prediction sets* that provably contain the correct answer with high probability (e.g., 95%), while only relying on a few in-task examples. Specifically, we focus on conformal prediction (CP)—a model-agnostic and distribution-free methodology

for creating confidence-based set predictions (Vovk et al., 2005). Concretely, suppose we have been given  $n$  examples,  $(X_j, Y_j) \in \mathcal{X} \times \mathcal{Y}$ ,  $j = 1, \dots, n$ , as training data, that have been drawn exchangeably from some underlying distribution  $P$ . Let  $X_{n+1} \in \mathcal{X}$  be a new exchangeable test example for which we would like to predict  $Y_{n+1} \in \mathcal{Y}$ . The aim of conformal prediction is to construct a set-valued output,  $\mathcal{C}_\epsilon(X_{n+1})$ , that contains  $Y_{n+1}$  with *distribution-free marginal coverage* at a significance level  $\epsilon \in (0, 1)$ , i.e.,

$$\mathbb{P}(Y_{n+1} \in \mathcal{C}_\epsilon(X_{n+1})) \geq 1 - \epsilon. \quad (1)$$

A conformal model is considered to be *valid* if the frequency of error,  $Y_{n+1} \notin \mathcal{C}_\epsilon(X_{n+1})$ , does not exceed  $\epsilon$ . The challenge for few-shot learning, however, is that as  $n \rightarrow 0$ , standard CP methods quickly result in outputs  $\mathcal{C}_\epsilon(X_{n+1})$  so large that they lose all utility (e.g., a trivially valid classifier that returns all of  $\mathcal{Y}$ ). A conformal model is only considered to be *efficient* if  $\mathbb{E}[|\mathcal{C}_\epsilon(X_{n+1})|]$  is relatively small.

In this work, we approach this frustrating data sparsity issue by casting conformal prediction as a meta-learning paradigm over exchangeable collections of tasks. By being exposed to a set of similar, auxiliary tasks, our model can *learn to learn quickly* on the target task at hand. As a result, we can increase the data efficiency of our procedure, and are able to produce more precise—and confident—outputs.

Specifically, we use the auxiliary tasks to meta-learn both **a few-shot model and a quantile predictor**. The few-shot model provides relevance scores (i.e., *nonconformity scores*, see §3.1) for each possible label candidate  $y \in \mathcal{Y}$ , and the quantile predictor provides a threshold rule for including the candidate  $y$  in the prediction set,  $\mathcal{C}_\epsilon(X_{n+1})$ , or not. A good few-shot model should provide scores that clearly separate *correct* labels from *incorrect* labels—much like a maximum-margin model. Meanwhile, a good quantile predictor—which is intrinsically linked to the specific few-shot model used—should quantify what few-shot scores correspond to relatively “high” or relatively “low” values for that task (i.e., as the name suggests, they infer the target quantile of the expected distribution of few-shot scores). Both of these models must be able to operate effectively given only a few examples from the target task, hence how they are meta-learned over auxiliary tasks becomes crucial.

Consider the example of image classification for novel cate-

---

<sup>1</sup>Computer Science and Artificial Intelligence Laboratory, Massachusetts Institute of Technology, Cambridge, MA, USA. Correspondence to: Adam Fisch <fisch@csail.mit.edu>.

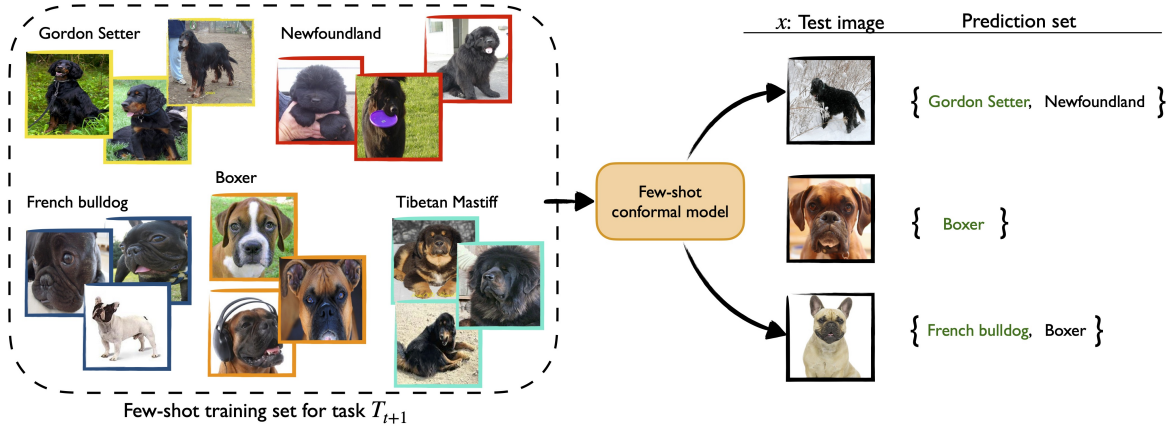


Figure 1. A demonstration of our conformalized few-shot learning procedure. Given a base model (e.g., a prototypical network for classification tasks (Snell et al., 2017)) and a few demonstrations of a new task, our method produces a prediction *set* that carries desirable guarantees that it contains the correct answer with high probability. Like other meta-learning algorithms, our approach leverages information gained from  $t$  other, similar tasks—here to make more precise and confident predictions on the new task,  $T_{t+1}$ .

gories (see Figure 1 for an illustration). The goal is to predict the class of a new test image out of several never-before-seen categories—while only given a handful of training examples per category. In terms of auxiliary tasks, we are given access to similarly-framed image classification tasks (e.g., *cat* classes instead of *dog* classes as in Figure 1). In this case, we can compute relevance by using a prototypical network (Snell et al., 2017) to measure the Euclidean distance between the test image’s representation and the average representation of the considered candidate class’s support images (i.e., prototype). Our quantile predictor then computes a “distance cut-off” that represents the largest distance between a label prototype and the test example that just covers the desired percentage of correct labels. Informally, on the auxiliary tasks, the prototypical network will learn efficient features, while the quantile predictor will learn what typically constitutes expected prototypical distances for correct labels when using the trained network.

We demonstrate that these two meta-learned components combine to make an efficient and simple-yet-effective approach to few-shot conformal prediction, all while retaining desirable theoretical performance guarantees. We empirically validate our approach on image classification, relation classification for textual entities, and chemical property prediction for drug discovery. Our code is publicly available.<sup>1</sup>

In summary, our main contributions are as follows:

- A novel theoretical extension of conformal prediction to include few-shot prediction with auxiliary tasks
- A principled meta-learning framework for constructing confident set-valued classifiers for new target tasks
- A demonstration of the practical utility of our framework

<sup>1</sup><https://github.com/ajfishch/few-shot-cp>.

across a range of classification and regression tasks.

## 2 Related Work

**Uncertainty estimation.** In recent years, there has been a growing research interest in estimating uncertainty in model predictions. A large amount of work has been dedicated towards *calibrating* the model posterior,  $p_{\theta}(\hat{y}_{n+1}|x_{n+1})$ , such that the true accuracy,  $y_{n+1} = \hat{y}_{n+1}$ , is indeed equal to the estimated probability (Niculescu-Mizil & Caruana, 2005; Lakshminarayanan et al., 2017; Lee et al., 2018). In theory, these estimates could be used to create confident prediction sets  $\mathcal{C}_{\epsilon}(X_{n+1})$ . Unlike CP, however, these methods are not guaranteed to be accurate, and often suffer from miscalibration in practice—and this is especially true for modern neural networks (Guo et al., 2017; Ashukha et al., 2020; Hirschfeld et al., 2020). In a similar vein, Bayesian formalisms underlie several popular approaches to quantifying predictive uncertainty via computing the posterior distribution over model parameters (Neal, 1996; Graves, 2011; Hernández-Lobato & Adams, 2015; Gal & Ghahramani, 2016). The quality of these methods, however, largely hinges on both (1) the degree of approximation required in computing the posterior, and (2) the suitability, or “correctness”, of the presumed prior distribution.

**Conformal prediction.** As introduced in §1, conformal prediction (Vovk et al., 2005) provides a model-agnostic and finite-sample, distribution-free method for obtaining prediction sets with marginal coverage guarantees. Most pertinent to our work, Linusson et al. (2014) carefully analyze the effects of calibration set size on CP performance. For precise prediction sets, they recommend using at least a few hundred examples for calibration—much larger than the few-shot settings considered here. When the amount of available data is severely restricted, the predicted sets typically become unusably large. Johansson et al. (2015) and

Carlsson et al. (2015) introduce similarly motivated approximations to CP with small calibration sets via interpolating calibration instances or using modified  $p$ -value definitions, respectively. Both methods are heuristics, however, and fail to provide finite-sample guarantees. Our work also complements several recent directions that explore conformal prediction in the context of various validity conditions, such as conditional, risk-controlling, admissible, or equalized coverage (Chernozhukov et al., 2019; Cauchois et al., 2020; Kivaranovic et al., 2020; Romano et al., 2019; 2020; Bates et al., 2020; Fisch et al., 2021, *inter alia*).

**Few-shot learning.** Despite the many successes of machine learning models, learning from limited data is still a significant challenge (Bottou & Bousquet, 2008; Lake et al., 2015; Wang et al., 2020). Our work builds upon the extensive few-shot learning literature by introducing a principled way of obtaining confidence intervals via meta-learning. Meta-learning has become a popular approach to transferring knowledge gained from auxiliary tasks—e.g., via featurizations or statistics (Edwards & Storkey, 2017)—to a target task that is otherwise resource-limited (Vinyals et al., 2016; Finn et al., 2017; Snell et al., 2017; Bertinetto et al., 2019; Bao et al., 2020). We leverage the developments in this area for our models (see Appendix B.1).

### 3 Background

We begin with a review of conformal prediction (see Shafer & Vovk, 2008). Here, and in the rest of the paper, upper-case letters ( $X$ ) denote random variables; lower-case letters ( $x$ ) denote scalars, and script letters ( $\mathcal{X}$ ) denote sets, unless otherwise specified. A list of notation definitions is given in Table A.1. All proofs are deferred to Appendix A.

#### 3.1 Nonconformity measures

Given a new example  $x$ , for every candidate label  $y \in \mathcal{Y}$ , conformal prediction applies a simple test to either accept or reject the null hypothesis that the pairing  $(x, y)$  is correct. The test statistic for this hypothesis test is a *nonconformity measure*,  $\mathcal{S}((x, y), \mathcal{D})$ , where  $\mathcal{D}$  is a dataset of exchangeable, correctly labeled examples. Informally, a lower value of  $\mathcal{S}$  reflects that point  $(x, y)$  “conforms” to  $\mathcal{D}$ , whereas a higher value of  $\mathcal{S}$  reflects that  $(x, y)$  is atypical relative to  $\mathcal{D}$ . A practical choice for  $\mathcal{S}$  is model-based likelihood, e.g.,  $-\log p_\theta(y|x)$ , where  $\theta$  is a model fit to  $\mathcal{D}$  using some learning algorithm  $\mathcal{A}$  (such as gradient descent). It is also important that  $\mathcal{S}$  preserves exchangeability of its inputs. Let  $Z_j := (X_j, Y_j)$ ,  $j = 1, \dots, n$  be the training data. Then, for test point  $x \in \mathcal{X}$  and candidate label  $y \in \mathcal{Y}$ , we calculate the *nonconformity scores* for  $(x, y)$  as:

$$\begin{aligned} V_j^{(x,y)} &:= \mathcal{S}(Z_j, Z_{1:n} \cup \{(x, y)\}), \\ V_{n+1}^{(x,y)} &:= \mathcal{S}((x, y), Z_{1:n} \cup \{(x, y)\}). \end{aligned} \quad (2)$$

Note that this formulation, referred to as *full* conformal prediction, requires running the learning algorithm  $\mathcal{A}$  that underlies  $\mathcal{S}$  potentially many times for every new test point (i.e.,  $|\mathcal{Y}|$  times). “Split” conformal prediction (Papadopoulos, 2008)—which uses a held-out training set to learn  $\mathcal{S}$ , and therefore also preserves exchangeability—is a more computationally attractive alternative, but comes at the expense of predictive efficiency when data is limited.<sup>2</sup>

#### 3.2 Conformal prediction

To construct the final *prediction* for the new test point  $x$ , the classifier tests the nonconformity score for each label  $y$ ,  $V_{n+1}^{(x,y)}$ , against a desired significance level  $\epsilon$ , and includes all  $y$  for which the null hypothesis—that the candidate pair  $(x, y)$  is *conformal*—is not rejected. This is achieved by comparing the nonconformity score of the test candidate to the scores computed over the first  $n$  labeled examples. This comparison leverages the quantile function, where for a random variable  $V$  sampled from distribution  $F$  we define

$$\text{Quantile}(\beta; F) := \inf\{v : F(v) \geq \beta\}. \quad (3)$$

In our case,  $F$  is the distribution over the  $n + 1$  nonconformity scores, denoted  $V_{1:n+1}$ . However, as we do not know  $V_{n+1}^{(x,y)}$  for the true  $y^*$ , we use an “inflated” quantile:

**Lemma 3.1** (Inflated quantile). *Assume that  $V_j$ ,  $j = 1, \dots, n + 1$  are exchangeable random variables. Then for any  $\beta \in (0, 1)$ ,  $\mathbb{P}(V_{n+1} \leq \text{Quantile}(\beta, V_{1:n} \cup \{\infty\})) \geq \beta$ .*

Conformal prediction then guarantees marginal coverage by including all labels  $y$  for which  $V_{n+1}^{(x,y)}$  is below the inflated quantile of the  $n$  training points, as summarized:

**Theorem 3.2** (CP, Vovk et al. (2005)). *Assume that examples  $(X_j, Y_j)$ ,  $j = 1, \dots, n + 1$  are exchangeable. For any nonconformity measure  $\mathcal{S}$  and  $\epsilon \in (0, 1)$ , define the conformal set (based on the first  $n$  examples) at  $x \in \mathcal{X}$  as*

$$\mathcal{C}_\epsilon(x) := \left\{ y \in \mathcal{Y} : V_{n+1}^{(x,y)} \leq \text{Quantile}(1 - \epsilon; V_{1:n}^{(x,y)} \cup \{\infty\}) \right\}.$$

Then  $\mathcal{C}_\epsilon(X_{n+1})$  satisfies Eq. (1).

Though Theorem 3.2 provides guarantees for any training set size  $n$ , in practice  $n$  must be fairly large (e.g., 1000) to achieve reasonable, stable performance—in the sense that  $\mathcal{C}_\epsilon$  will not be too large on average (Lei et al., 2018; Bates et al., 2020). This is a key hurdle for few-shot conformal prediction, where  $n = k$  is assumed to be small (e.g., 16).

<sup>2</sup> Split conformal prediction also allows for simple nonconformity score calculations for regression tasks. For example, assume that a training set has been used to train a fixed regression model,  $f_\theta(x)$ . The absolute error nonconformity measure,  $|y - f_\theta(x)|$ , can then be easily evaluated for all  $y \in \mathbb{R}$ . Furthermore, as the absolute error monotonically increases away from  $f_\theta(x)$ , the conformal prediction  $\mathcal{C}_\epsilon$  simplifies to a closed-form *interval*.

## 4 Few-shot Meta Conformal Prediction

We now propose a general meta-learning paradigm for training efficient conformal predictors, while relying only on a very limited number of *in-task* examples.

At a high level, like other meta-learning algorithms, our approach leverages information gained from  $t$  other, similar tasks in order to perform better on task  $t + 1$ . In our setting we achieve this by learning a more statistically powerful nonconformity measure and quantile estimator than would otherwise be possible using only the limited data available for the target task. Our method uses the following recipe:

1. We meta-learn (and calibrate) a nonconformity measure and quantile predictor over a set of auxiliary tasks;
2. We adapt our meta nonconformity measure and quantile predictor using the examples we have for our target task;
3. We compute a conformal prediction set for a new input  $x \in \mathcal{X}$  by including all labels  $y \in \mathcal{Y}$  whose meta-learned nonconformity score is below the predicted  $1 - \epsilon$  quantile.

Pseudo-code for our meta CP procedure is given in Algorithm 1. This skeleton focuses on classification; regression follows similarly. Our framework is model agnostic, in that it allows for practically any meta-learning implementation for both nonconformity and quantile prediction models.

In the following sections, we break down our approach in detail. In §4.1 we precisely formulate our few-shot learning setup with auxiliary tasks. In §4.2 and §4.3 we describe our meta-learning and meta-calibration setups, respectively. Finally, in §4.4 we discuss further theoretical extensions. For a complete technical description of our modeling choices and training strategy for our experiments, see Appendix B.

### 4.1 Task formulation

In this work, we assume access to  $t$  auxiliary tasks,  $T_i$ ,  $i = 1, \dots, t$ , that we wish to leverage to produce tighter uncertainty sets for predictions on a new task,  $T_{t+1}$ . Furthermore, we assume that these  $t + 1$  tasks are *exchangeable* with respect to some task distribution,  $P_{\mathcal{T}}$ . Here, we treat  $P_{\mathcal{T}}$  as a distribution over random distributions, where each task  $T_i \in \mathcal{T}$  defines a task-specific distribution,  $P_{XY} \sim P_{\mathcal{T}}$ , over examples  $(X, Y) \in \mathcal{X} \times \mathcal{Y}$ . The randomness is in both the task's relation between  $X$  and  $Y$ , and the task's data.

For each of the  $t$  auxiliary tasks, we do not make any assumptions on the amount of data we have (though, in general, we expect them to be relatively unrestricted). On the new task  $T_{t+1}$ , however, we only assume a total of  $k$  training examples. Our goal is then to develop a task-agnostic uncertainty estimation strategy that generalizes well to new examples

---

### Algorithm 1 Meta conformal prediction with auxiliary tasks.

---

**Definitions:**  $T_{1:t+1}$  are exchangeable tasks.  $\mathcal{I}_{\text{train}} \cup \mathcal{I}_{\text{cal}}$  are the  $t$  tasks used for meta-training and meta-calibration.  $z_{1:k} \in (\mathcal{X} \times \mathcal{Y})^k$  are the  $k$  support examples for target task  $T_{t+1}$ .  $x \in \mathcal{X}$  is the target task input.  $\mathcal{Y}$  is the label space.  $\epsilon$  is the significance.

```

1: function PREDICT( $x, z_{1:k}, T_{1:t}, \epsilon$ )
2:   # Learn  $\hat{S}$  and  $\hat{P}$  on meta-training tasks (§4.2).
3:   #  $\hat{S}$  and  $\hat{P}$  are meta nonconformity/quantile models.
4:    $\hat{S}, \hat{P}_{1-\epsilon} \leftarrow \text{TRAIN}(T_i, i \in \mathcal{I}_{\text{train}})$ 
5:   # Predict the  $1 - \epsilon$  quantile.
6:    $\hat{Q}_{t+1} \leftarrow \hat{P}_{1-\epsilon}(z_{1:k}; \phi_{\text{meta}})$ 
7:   # Initialize empty output set.
8:    $\mathcal{M}_{\epsilon} \leftarrow \{\}$ 
9:   # (Note that for regression tasks, where  $|\mathcal{Y}| = \infty$ , for
10:    certain  $\hat{S}$  the following simplifies to a closed-form
11:    # interval, making it tractable—see §3.1, footnote 2.)
12:  for  $y \in \mathcal{Y}$  do
13:    # Compute the nonconformity score for label  $y$ .
14:     $\hat{V}_{t+1,k+1}^{(x,y)} \leftarrow \hat{S}((x, y), z_{1:k}; \theta_{\text{meta}})$ 
15:    # Compare to the calibrated quantile (§4.3).
16:    if  $\hat{V}_{t+1,k+1}^{(x,y)} \leq \hat{Q}_{t+1} + \Lambda(1 - \epsilon, \mathcal{I}_{\text{cal}})$  then
17:       $\mathcal{M}_{\epsilon} \leftarrow \mathcal{M}_{\epsilon} \cup \{y\}$ 
18:  return  $\mathcal{M}_{\epsilon}$ 
    
```

---

from the task's unseen test set,  $(X_{t+1}^{\text{test}}, Y_{t+1}^{\text{test}})$ .<sup>3</sup> Specifically, we desire finite-sample marginal *task* coverage, as follows:

**Definition 4.1** (Task validity). Let  $\mathcal{M}_{\epsilon}$  be a set-valued predictor.  $\mathcal{M}_{\epsilon}$  is considered to be *valid* across tasks if for any task distribution  $P_{\mathcal{T}}$  and  $\epsilon \in (0, 1)$ , we have

$$\mathbb{P}\left(Y_{t+1}^{\text{test}} \in \mathcal{M}_{\epsilon}(X_{t+1}^{\text{test}})\right) \geq 1 - \epsilon. \quad (4)$$

Note that we require the marginal coverage guarantee above to hold on average across tasks and their examples.

### 4.2 Meta-learning conformal prediction models

Given our collection of auxiliary tasks, we would like to meta-learn both (1) an effective nonconformity measure that is able to adapt quickly to a new task using only  $k$  examples; and (2) a quantile predictor that is able to robustly identify the  $1 - \epsilon$  quantile of that same meta nonconformity measure, while only using the same  $k$  examples.

Prior to running our meta-learning algorithm of choice, we split our set of  $t$  auxiliary tasks into disjoint sets of training tasks,  $\mathcal{I}_{\text{train}}$ , and calibration tasks,  $\mathcal{I}_{\text{cal}}$ , where  $|\mathcal{I}_{\text{train}}| + |\mathcal{I}_{\text{cal}}| = t$ . See Table 1 for an overview of the different splits. We use  $\mathcal{I}_{\text{train}}$  to learn our meta nonconformity measures and quantile predictors, which we discuss now. Additional technical details are contained in Appendix B.1.

<sup>3</sup>For ease of notation, we write  $X_{t+1}^{\text{test}}$  to denote the  $(k + 1)$ th example of task  $T_{t+1}$ , i.e., the new test point after observing  $k$  training points. This is equivalent to test point  $X_{n+1}$  from §3.



Auxiliary {	Task Split	# Tasks	# Examples / Task
	Meta-training	$ \mathcal{I}_{\text{train}} $	$\gg k$
	Meta-calibration	$ \mathcal{I}_{\text{cal}} $	$k + m_i$
}	Test	1	$k$

Table 1. An overview of the data assumptions for a single test task “episode”. We use  $|\mathcal{I}_{\text{train}}| + |\mathcal{I}_{\text{cal}}| = t$  total auxiliary tasks to create more precise uncertainty estimates for the  $(t + 1)$ th test task. This is repeated for each test task (§5).  $m_i \gg k$  is the number of extra examples per calibration task that are used to compute an empirical CDF when finding  $\Lambda(\beta; \mathcal{I}_{\text{cal}})$ —it may vary per task.

**Meta nonconformity measure.** Let  $\hat{\mathcal{S}}((x, y), \mathcal{D}; \theta_{\text{meta}})$  be a *meta nonconformity measure*, where  $\theta_{\text{meta}}$  are meta parameters learned over the auxiliary tasks in  $\mathcal{I}_{\text{train}}$ . Since  $\theta_{\text{meta}}$  is fixed after the meta training period,  $\hat{\mathcal{S}}$  preserves exchangeability over new collections of exchangeable tasks (i.e.,  $\mathcal{I}_{\text{cal}}$ ) and task examples. Let  $Z_{i,j} := (X_{i,j}, Y_{i,j})$ ,  $j = 1, \dots, k$  be the few-shot training data for a task  $T_i$  (here  $i$  is the task index, while  $j$  is the example index). Given a new test point  $x \in \mathcal{X}$  and candidate pairing  $(x, y)$ , the *meta nonconformity scores* for  $(x, y)$  are

$$\begin{aligned} \hat{V}_{i,j}^{(x,y)} &:= \hat{\mathcal{S}}(Z_{i,j}, Z_{i,1:k} \cup \{(x, y)\}; \theta_{\text{meta}}), \\ \hat{V}_{i,k+1}^{(x,y)} &:= \hat{\mathcal{S}}((x, y), Z_{i,1:k} \cup \{(x, y)\}; \theta_{\text{meta}}). \end{aligned} \quad (5)$$

As an example, Figure 2 demonstrates how we compute  $\hat{V}_{i,k+1}^{(x,y)}$  using the distances from a meta-learned prototypical network following the setting in Figure 1.

Computing all  $k + 1$  scores  $|\mathcal{Y}|$  times is typically tractable due to the few number of examples (e.g.,  $k \approx 16$ ) and the underlying properties of the meta-learning algorithm driving  $\hat{\mathcal{S}}$ . For example, prototypical networks only require a forward pass. A naive approach to few-shot conformal prediction is to exploit this efficiency, and simply run full CP using all  $k + 1$  data points. Nevertheless, though a strong baseline, using only  $k + 1$  points to compute an empirical quantile is still suboptimal. As we discuss next, instead we choose to regress the desired quantile directly from  $Z_{i,1:k}$ , and disregard the empirical quantile completely.

Since we predict the quantile instead of relying on the empirical quantile, we do not have to retain exchangeability for  $Z_{i,1:k}$ . As a result, we switch to “split” CP (§3.1), and *do not* include  $(x, y)$  when calculating  $\hat{V}_{i,j}^{(x,y)}$ , as this is faster.

**Meta quantile predictor.** Let  $\hat{\mathcal{P}}_{\beta}(\mathcal{D}; \phi_{\text{meta}})$  be a *meta  $\beta$ -quantile predictor*, where  $\phi_{\text{meta}}$  are the meta parameters learned over the auxiliary tasks in  $\mathcal{I}_{\text{train}}$ .  $\hat{\mathcal{P}}_{\beta}$  is trained to predict the  $\beta$ -quantile of  $F$ —where  $F$  is the underlying task-specific distribution of nonconformity scores—given  $\mathcal{D}$ , a dataset of  $Z = (X, Y)$  pairs sampled from that task.

As some intuition for this approach, recall that in calculating

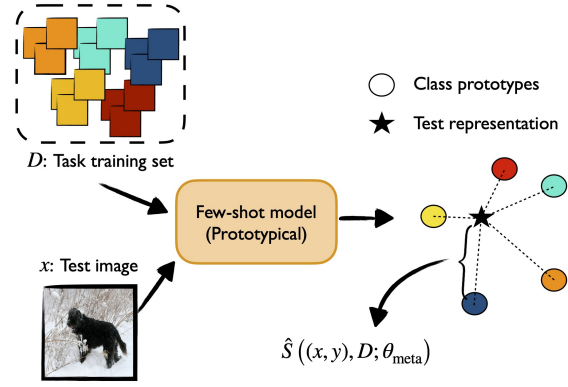


Figure 2. An example of using a prototypical network (Snell et al., 2017) to compute meta nonconformity scores. If  $\hat{\mathcal{S}}$  is well-trained, the distance between the test point and the correct class prototype should be small, and the distance to incorrect prototypes large, even when the number of in-task training examples is limited.

Quantile( $\beta; F$ ) given exchangeable samples  $v_{1:n} \sim F$ , we implicitly need to estimate  $\mathbb{P}(V_{n+1} \leq v \mid v_{1:n})$ . For an appropriate parametrization  $\psi$  of  $F$ , de Finetti’s theorem for exchangeable sequences allows us to write

$$\mathbb{P}(V_{n+1} \leq v \mid v_{1:n}) \propto \int_{-\infty}^v \int_{\Psi} p(v \mid \psi) \prod_{i=1}^n p(v_i \mid \psi) p(\psi) d\psi dv.$$

In this sense, meta-learning over auxiliary task distributions may help us learn a better prior over latent parametrizations  $\psi$ —which in turn may help us better model the  $\beta$ -quantile than we could have, given only  $k$  samples and nothing else.

We develop a simple approach to modeling and learning  $\hat{\mathcal{P}}_{\beta}$ . Given the training examples  $Z_{i,1:k}$ , we use a deep sets model (Zaheer et al., 2017) parameterized by  $\phi_{\text{meta}}$  to predict the  $\beta$ -quantile of  $\hat{V}_{i,k+1}^{\text{test}}$ , the random variable representing the nonconformity score of the test point,  $Z_{i,k+1} := (X_{i,k+1}, Y_{i,k+1})$ . We optimize  $\phi_{\text{meta}}$  as

$$\min_{\phi} \sum_{i \in \mathcal{I}_{\text{train}}} \left( \hat{\mathcal{P}}_{\beta}(Z_{i,1:k}; \phi) - \text{Quantile}(\beta; \hat{V}_{i,k+1}^{\text{test}}) \right)^2, \quad (6)$$

where we estimate the target,  $\text{Quantile}(\beta; \hat{V}_{i,k+1}^{\text{test}})$ , using  $m \gg k$  extra examples sampled from the training task.

In practice, we found that choosing to first transform  $Z_{i,1:k}$  to *leave-one-out* meta nonconformity scores,

$$\hat{L}_{i,j} := \hat{\mathcal{S}}(Z_{i,j}, Z_{i,1:k} \setminus Z_{i,j}; \theta_{\text{meta}}), \quad (7)$$

and providing  $\hat{\mathcal{P}}_{\beta}$  with these scalar leave-one-out scores as inputs, performs reasonably well and is lightweight to implement. Inference using  $\hat{\mathcal{P}}_{\beta}$  is illustrated in Figure 3.

**Training strategy.** The meta nonconformity measure  $\hat{\mathcal{S}}$  and meta quantile predictor  $\hat{\mathcal{P}}_{\beta}$  are tightly coupled, as given a fixed  $\hat{\mathcal{S}}$ ,  $\hat{\mathcal{P}}_{\beta}$  learns to model its behavior on new data. A

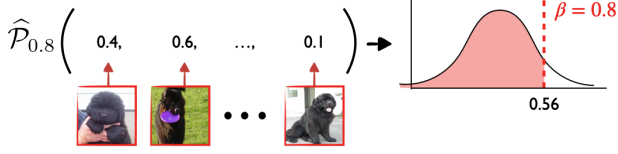


Figure 3. An illustration of using our meta-learned quantile predictor  $\hat{\mathcal{P}}_\beta$  to infer the  $\beta$ -quantile of the distribution of  $\hat{V}_{i,k+1}^{\text{test}}$ , given the few examples from  $T_i$ 's training set. The numbers above each image reflect the leave-one-out scores we use as inputs, see Eq. (7).

straightforward, but data inefficient, approach to training  $\hat{\mathcal{S}}$  and  $\hat{\mathcal{P}}_\beta$  is to split the collection of auxiliary tasks in  $\mathcal{I}_{\text{train}}$  in two, i.e.,  $\mathcal{I}_{\text{train}} = \mathcal{I}_{\text{train}}^{(1)} \cup \mathcal{I}_{\text{train}}^{(2)}$ , and then train  $\hat{\mathcal{S}}$  on  $\mathcal{I}_{\text{train}}^{(1)}$ , followed by training  $\hat{\mathcal{P}}_\beta$  on  $\hat{\mathcal{S}}$ 's predictions over  $\mathcal{I}_{\text{train}}^{(2)}$ . The downside of this strategy is that both  $\hat{\mathcal{S}}$  and  $\hat{\mathcal{P}}_\beta$  may be sub-optimal, as neither can take advantage of all of  $\mathcal{I}_{\text{train}}$ .

We employ a slightly more involved, but more data efficient approach, where we split  $\mathcal{I}_{\text{train}}$  into  $k_f$  folds, i.e.,  $\mathcal{I}_{\text{train}} = \bigcup_{f=1}^{k_f} \mathcal{I}_{\text{train}}^{(f)}$ . We then train  $k_f$  separate meta nonconformity measures  $\hat{\mathcal{S}}_f$ , where we leave out fold  $f$  from the training data. Using  $\hat{\mathcal{S}}_f$ , we compute nonconformity scores on fold  $f$ 's data, aggregate these nonconformity scores across all  $k_f$  folds, and train the meta quantile predictor on this union. Finally, we train another nonconformity measure on *all* of  $\mathcal{I}_{\text{train}}$ , which we use as our ultimate  $\hat{\mathcal{S}}$ . This way we are able to use all of  $\mathcal{I}_{\text{train}}$  for training both  $\hat{\mathcal{S}}$  and  $\hat{\mathcal{P}}_\beta$ . This process is illustrated in Figure B.1. Note that it is not problematic for  $\hat{\mathcal{P}}_\beta$  to be trained on the collection of  $\hat{\mathcal{S}}$  instances trained on  $k_f - 1$  folds, but then later used to model one  $\hat{\mathcal{S}}$  trained on *all* the data, since it will be calibrated (next, in §4.3).

### 4.3 Calibrating meta-learned conformal prediction

Though  $\hat{\mathcal{P}}_\beta$  may obtain low empirical error after training, it does not have any inherent rigorous guarantees out-of-the-box. Given our held-out set of auxiliary tasks  $\mathcal{I}_{\text{cal}}$ , however, we can quantify the uncertainty in  $\hat{\mathcal{P}}_\beta$  (i.e., how far off it may be from the true quantile), and calibrate it accordingly. The following lemma formalizes our meta calibration procedure:

**Lemma 4.2** (Meta calibration). *Assume  $\hat{Q}_i$ ,  $i \in \mathcal{I}_{\text{cal}}$  are the (exchangeable) meta  $\beta$ -quantile predictions produced by  $\hat{\mathcal{P}}_\beta$  for tasks  $T_i$ ,  $i \in \mathcal{I}_{\text{cal}}$ . Let  $\hat{V}_{i,k+1}^{\text{test}}$  be the meta nonconformity score for a new sample from task  $T_i$ , where  $F_i$  is its distribution function. Define the correction  $\Lambda(\beta; \mathcal{I}_{\text{cal}})$  as*

$$\Lambda(\beta; \mathcal{I}_{\text{cal}}) := \inf \left\{ \lambda: \frac{1}{|\mathcal{I}_{\text{cal}}| + 1} \sum_{i \in \mathcal{I}_{\text{cal}}} F_i(\hat{V}_{i,k+1}^{\text{test}} \leq \hat{Q}_i + \lambda) \geq \beta \right\}.$$

We then have that  $\mathbb{P}(\hat{V}_{t+1,k+1}^{\text{test}} \leq \hat{Q}_{t+1} + \Lambda(\beta; \mathcal{I}_{\text{cal}})) \geq \beta$ .

It is important to pause to clarify at this point that calculating  $\Lambda(\beta; \mathcal{I}_{\text{cal}})$  requires knowledge of the true meta nonconformity distribution functions,  $F_i$ , for all calibration tasks. For

simplicity, we write Lemma 4.2 and the following Theorem 4.3 as if these distribution functions are indeed known (again, only for calibration tasks). In practice, however, we typically only have access to an *empirical* distribution function over  $m_i$  task samples. In this case, Lemma 4.2 holds in expectation over task samples  $Z_{i,k:k+m_i}$ , as for an empirical distribution function of  $m$  points,  $\hat{F}_m$ , we have  $\mathbb{E}[\hat{F}_m] = F$ . Furthermore, for large enough  $m_i$ , concentration results suggest that we can approximate  $F_i$  with little error given a particular sample (this is the focus of §4.4).

That said, in a nutshell, Lemma 4.2 allows us to probabilistically adjust for the error in  $\hat{\mathcal{P}}_\beta$ , such that it is guaranteed to produce valid  $\beta$ -quantiles on average. We can then perform conformal inference on the target task by comparing each meta nonconformity score for a point  $x \in \mathcal{X}$  and candidate label  $y \in \mathcal{Y}$  to the calibrated meta quantile, and keep all candidates with nonconformity scores that fall below it.

**Theorem 4.3** (Meta CP). *Assume that tasks  $T_i$ ,  $i \in \mathcal{I}_{\text{cal}}$  and  $T_{t+1}$  are exchangeable, and that their nonconformity distribution functions  $F_i$  are known. For any meta quantile predictor  $\hat{\mathcal{P}}_{1-\epsilon}$ , meta nonconformity measure  $\hat{\mathcal{S}}$ , and  $\epsilon \in (0, 1)$ , define the meta conformal set (based on the tasks in  $\mathcal{I}_{\text{cal}}$  and the  $k$  training examples of task  $T_{t+1}$ ) at  $x \in \mathcal{X}$  as*

$$\mathcal{M}_\epsilon(x) := \left\{ y \in \mathcal{Y}: \hat{V}_{t+1,k+1}^{(x,y)} \leq \hat{Q}_{t+1} + \Lambda(1-\epsilon; \mathcal{I}_{\text{cal}}) \right\},$$

where  $\hat{Q}_{t+1}$  is the result of running  $\hat{\mathcal{P}}_{1-\epsilon}$  on the  $k$  training examples of task  $T_{t+1}$ . Then  $\mathcal{M}_\epsilon(X_{t+1}^{\text{test}})$  satisfies Eq. (4).

It should be acknowledged that Theorem 4.3 guarantees coverage *marginally* over tasks, as specified in Eq. (4). Given appropriate assumptions on the quantile predictor  $\hat{\mathcal{P}}_{1-\epsilon}$ , we can achieve task-conditional coverage asymptotically:

**Definition 4.4** (Consistency). We say  $\hat{\mathcal{P}}_{1-\epsilon}$  is an asymptotically consistent estimator of the  $1 - \epsilon$  quantile if

$$|\hat{\mathcal{P}}_{1-\epsilon}(Z_{i,1:k}; \phi_{\text{meta}}) - \text{Quantile}(1 - \epsilon, F_i)| = o_{\mathbb{P}}(1)$$

as  $k \rightarrow \infty$ , where  $F_i$  is the CDF of nonconformity scores for any task  $t_i \in \mathcal{T}$ . In other words,  $\hat{\mathcal{P}}_{1-\epsilon}$  converges in probability to the true quantile given enough in-task data.

**Proposition 4.5** (Asymptotic meta CP). *If  $\hat{\mathcal{P}}_{1-\epsilon}$  is asymptotically consistent, then as  $k \rightarrow \infty$  the meta conformal set  $\mathcal{M}_\epsilon$  achieves asymptotic conditional coverage, where*

$$\mathbf{1}\left\{ \mathbb{P}(Y_{t+1}^{\text{test}} \in \mathcal{M}_\epsilon(X_{t+1}^{\text{test}}) \mid T_{t+1} = t_{t+1}) \geq 1 - \epsilon \right\} = 1 - o_{\mathbb{P}}(1).$$

This result simply claims that as the number of in-task samples  $k$  increases, our meta CP will converge towards valid coverage for all tasks, not just on average. By itself, this is not particularly inspiring: after all, standard CP also becomes viable as  $k \rightarrow \infty$ . Rather, the key takeaway is that this desirable behavior is nicely preserved in our meta setup as well. In Figure 5 we demonstrate that our  $\hat{\mathcal{P}}_{1-\epsilon}$  indeed progresses towards task-conditional coverage as  $k$  grows.

#### 4.4 Meta-learned approximate conformal prediction

Recall that a key assumption in the theoretical results established in the previous section is that the distribution functions of our calibrations tasks,  $F_i$  where  $i \in \mathcal{I}_{\text{cal}}$ , are known. In this section we turn to analyze the (much more common) setting where these  $F_i$  must instead be estimated empirically. In this case, Theorem 4.3 holds in expectation over the samples chosen for the calibration tasks. Furthermore, standard concentration results suggest that we can approximate  $F_i$  with little error, given enough empirical samples (which, in general, we assume we have for our calibration tasks). We now further adapt Theorem 4.3 to be conditionally valid with respect to the labeled examples that are used when replacing each task  $F_i$  with its plug-in estimate,  $\hat{F}_{m_i}$ .

First, we formalize a PAC-type 2-parameter validity definition (similar to *training conditional CP* in Vovk (2012)):

**Definition 4.6** ( $(\delta, \epsilon)$  task validity).  $\mathcal{M}_\epsilon$  is  $(\delta, \epsilon)$  task valid if for any task distribution  $P_{\mathcal{T}}$ ,  $\epsilon \in (0, 1)$ , and  $\delta \in (0, 1)$ ,

$$\mathbb{P}\left(\mathbb{P}\left(Y_{t+1}^{\text{test}} \in \mathcal{M}_\epsilon(X_{t+1}^{\text{test}})\right) \geq 1 - \epsilon\right) \geq 1 - \delta. \quad (8)$$

The outer probability is taken with respect to the data samples used for calibration. The basic idea here is to include a secondary confidence level  $\delta$  that allows us to control how robust we are to sampling variance in our estimation of calibration tasks quantiles when computing  $\Lambda(\beta; \mathcal{I}_{\text{cal}})$ , our conformal prediction correction factor. We define a sample-conditional approach that is  $(\delta, \epsilon)$  task valid, as follows:

**Proposition 4.7** (Sample-conditional meta CP). Assume that all  $|\mathcal{I}_{\text{cal}}| = l$  calibration tasks are i.i.d., where for each task we have a fixed dataset that is also i.i.d. That is, for task  $T_i$ , we have drawn  $m_i$  i.i.d. training examples,  $(x_{i,j}, y_{i,j})$ ,  $j = 1, \dots, m_i$ . For any  $\delta \in (0, 1)$ ,  $\epsilon \in (0, 1)$ , and  $\alpha \in (0, 1 - (1 - \delta)^{\frac{1}{n}})$ , define the adjusted  $\epsilon'$  as

$$\epsilon' \leq \epsilon - \sqrt{\frac{-2}{l^2} \left( \sum_{i \in \mathcal{I}_{\text{cal}}} \gamma_i^2 \right) \log \left( 1 - \frac{1 - \delta}{(1 - \alpha)^l} \right)} \quad (9)$$

where  $\gamma_i = \sqrt{\frac{\log(2/\alpha)}{2m_i}}$ . Then  $\mathcal{M}_{\epsilon'}(X_{t+1}^{\text{test}})$  satisfies Eq. (8).

**Remark 4.8.** We are free to choose  $\alpha$  so as to optimize  $\epsilon'$ .

Increasing the number of auxiliary tasks or samples per task make  $\epsilon'$  closer to  $\epsilon$ . In §6 we show that we can achieve tight prediction sets in practice, even with small tolerances.

## 5 Experimental Setup

### 5.1 Evaluation tasks

**Image classification (CV).** As introduced in §1, the goal of few-shot image classification is to train a computer vision model that generalizes to entirely new image classes at

test time. We use the *miniImageNet* dataset (Vinyals et al., 2016), a downsampled version of a subset of classes from ImageNet (Deng et al., 2009). *miniImageNet* contains 100 classes that are divided into training, validation, and test class splits. Within each class partition, we construct  $K$ -shot  $N$ -way tasks, where  $K$  examples per class are used to discriminate between a sample of  $N$  distinct, novel classes. We use  $K = 16$  and  $N = 10$  in our experiments, for a total of  $k = 160$  training examples. In order to avoid label imbalanced accuracy, however, we choose to focus on Mondrian CP (Vovk et al., 2005), where validity is guaranteed across class type. Our meta nonconformity measure consists of a prototypical network on top of a CNN encoder.

**Relation classification (NLP).** Relation classification focuses on identifying the relationship between two entities mentioned in a given natural language sentence. In few-shot relation classification, the goal is to train an NLP model that generalizes to entirely new entity relationship types at test time. We use the FewRel 1.0 dataset (Han et al., 2018), which consists of 100 relations derived from 70k Wikipedia sentences. Like *miniImageNet*, the relation types are divided into training, validation, and test splits.<sup>4</sup> Within each partition, we sample  $K$ -shot  $N$ -way classification episodes (again with  $K = 16$  and  $N = 10$  and Mondrian CP, as in our CV task). Our meta nonconformity measure consists of a prototypical network on top of a CNN encoder with GloVe embeddings (Pennington et al., 2014).

**Chemical property prediction (Chem).** In-silico screening of chemical compounds is an important task for drug discovery. Given a new molecule, the goal is to predict its activity for a target chemical property. We use the ChEMBL dataset (Mayr et al., 2018), and regress the pChEMBL value (a normalized log-activity metric) for individual molecule-property pairs. We select a subset of 296 assays from ChEMBL, and divide them into training (208), validation (44), and test (44) splits. Within each partition, each assay’s pChEMBL values are treated as a regression task. We use  $k = 16$  training samples per task. Our meta nonconformity measure consists of a few-shot, closed-form ridge regressor (Bertinetto et al., 2019) on top of a directed Message Passing Network molecular encoder (Yang et al., 2019).<sup>5</sup>

### 5.2 Evaluation metrics

For each experiment, we use proper training, validation, and test meta-datasets of tasks. We use the meta-training tasks to learn all meta nonconformity measures  $\hat{\mathcal{S}}$  and meta quantile predictors  $\hat{\mathcal{P}}$ . We perform model selection for CP on the meta-validation tasks, and report final numbers on the meta-test tasks. For all methods, we report marginalized results over 5000 random trials, where in each trial we partition the

<sup>4</sup>We only use training/validation splits (the test set is hidden).

<sup>5</sup>We apply RRCM (Noureddinov et al., 2001) for full CP.

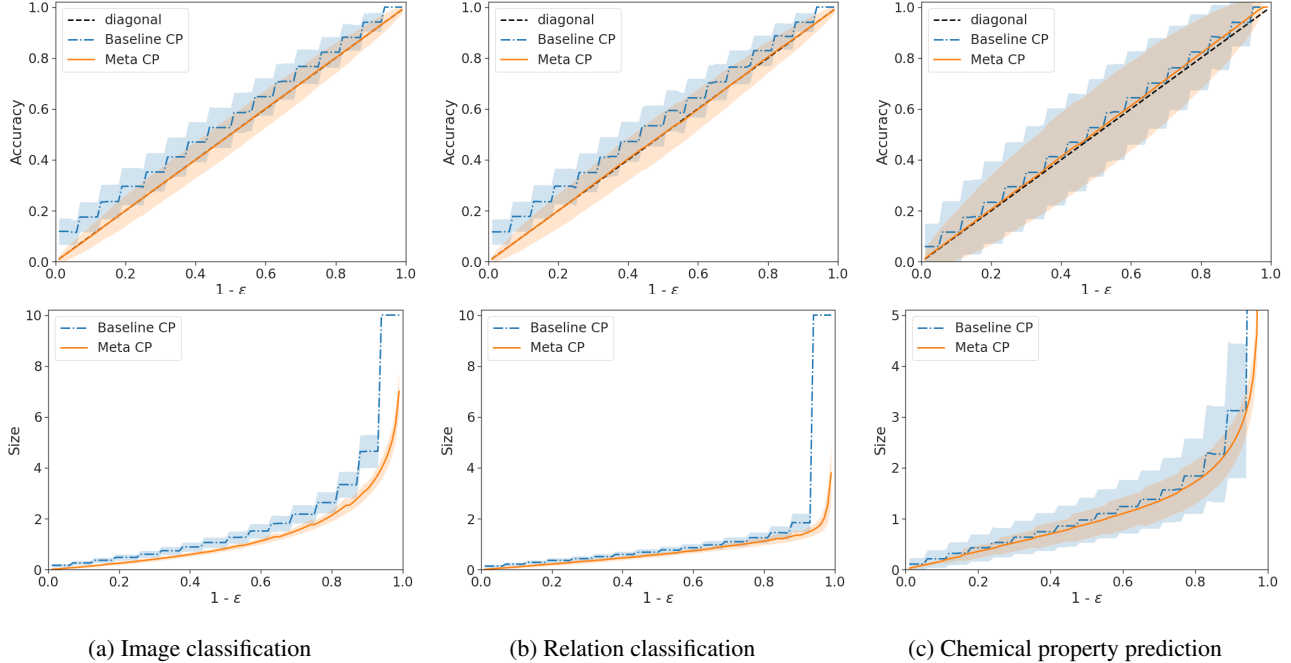


Figure 4. Few-shot CP results as a function of  $\epsilon$ . The size of the prediction set of our meta CP approach is significantly better (i.e., *smaller*) than that of our full CP baseline. Furthermore, our meta CP approach’s average accuracy level is close to the diagonal—allowing it to remain valid in the sense of Eq. (4), but also less conservative when making predictions. Note that we care more about the right-hand-side behavior of the above graphs (i.e., larger  $1 - \epsilon$ ), as they correspond to higher coverage guarantees.

data into  $l$  calibration tasks ( $T_{1:l}$ ) and one target task ( $T_{t+1}$ ). In all plots, shaded regions show  $\pm$  the standard deviation across trials. We use the following metrics:

**Prediction accuracy.** We measure accuracy as the rate at which the target label  $y \in \mathcal{Y}$  is contained within the predicted label set. For classification problems, the prediction is a discrete set, whereas in regression the prediction is a continuous interval. To be valid, a conformal model should have an average accuracy rate  $\geq 1 - \epsilon$ .

**Prediction size ( $\downarrow$ ).** We measure the average size of the output (i.e.,  $|\mathcal{C}_\epsilon|$ ) as a proxy for how *precise* the model’s predictions are. The goal is to make the prediction set as small as possible while still maintaining the desired accuracy.

### 5.3 Baselines

For all experiments, we compare our methods to full conformal prediction, in which we use a meta-learned non-conformity scores—as defined in Eq. (5). Though still a straightforward application of standard conformal calibration, meta-learning  $\hat{\mathcal{S}}$  with auxiliary tasks already adds significant statistical power to the model over an approach that would attempt to learn  $\mathcal{S}$  from scratch for each new task.

In addition to evaluating improvement over full CP, we compare our approach to other viable heuristics for making set valued predictions: Top- $k$  and Naive. In Top- $k$  we always take the  $k$ -highest ranked predictions. In Naive we

select likely labels until the cumulative softmax probability exceeds  $1 - \epsilon$ . While seeming related to our CP approach, we emphasize that these are only heuristics, and do not give the same theoretical performance guarantees.

## 6 Experimental Results

In the following, we present our main conformal few-shot results. We evaluate both our sample-conditional and unconditional meta conformal prediction approaches.

**Predictive efficiency.** We start by testing how our meta CP approach affects the size of the prediction set. Smaller prediction set sizes correspond to more efficient conformal models. We plot prediction set size as a function of  $\epsilon \in (0, 1)$  in Figure 4. Table 2 shows results for specific values of  $\epsilon$ , and also shows results for our *sample-conditional* meta CP approach, where we fix  $1 - \delta$  at 0.9 for all trials (note that the other meta results in Figure 4 and Table 2 are *unconditional*). Across all tasks and values of  $\epsilon$ , our meta CP performs the best in terms of efficiency. Moreover, the average size of the meta CP predictions increases smoothly as a function of  $\epsilon$ , while full CP suffers from discrete jumps in performance. Finally, we see that our sample-conditional ( $\delta = 0.1, \epsilon$ ) approach is only slightly more conservative than our unconditional meta CP method. This is especially true for domains with a higher number of auxiliary tasks and examples per auxiliary task (i.e., CV and NLP).



Task	Target Acc. ( $1 - \epsilon$ )	Baseline CP		Meta CP		$(\delta, \epsilon)$ -valid Meta CP	
		Acc.	$ \mathcal{C}_\epsilon $	Acc.	$ \mathcal{M}_\epsilon $	Acc.	$ \mathcal{M}_{k, \epsilon'} $
CV	0.95	1.00	10.00	0.95	3.80	0.96	3.98
	0.90	0.94	4.22	0.90	2.85	0.91	2.96
	0.80	0.83	2.38	0.80	1.89	0.81	1.95
	0.70	0.76	1.94	0.70	1.37	0.71	1.42
NLP	0.95	1.00	10.00	0.95	1.65	0.96	1.71
	0.90	0.94	1.84	0.90	1.39	0.91	1.42
	0.80	0.83	1.25	0.80	1.12	0.81	1.14
	0.70	0.76	1.10	0.70	0.93	0.71	0.94
Chem	0.95	1.00	inf	0.97	3.44	0.99	5.25
	0.90	0.94	3.28	0.92	2.62	0.95	3.02
	0.80	0.82	2.08	0.82	1.95	0.86	2.16
	0.70	0.71	1.59	0.72	1.56	0.76	1.70

Table 2. Few-shot CP results for  $\epsilon$  values. We report the empirical accuracy and raw prediction set size for our two meta CP methods, and compare to our baseline CP model (full CP with meta-learned  $\hat{S}$ ). For our sample-conditional meta CP approach, we fix  $\delta = 0.1$ . Note that CP can produce empty sets if no labels are deemed conformal, hence the average classification size may fall below 1 for high  $\epsilon$ .

Top-k: Size ( $k$ )	CV	NLP	Naive: Target Acc.	CV		NLP	
	Acc.	Acc.		Acc.	Size	Acc.	Size
5	0.96	0.99	0.95	0.97	4.38	0.99	2.98
3	0.88	0.98	0.90	0.94	3.50	0.99	2.45
1	0.60	0.79	0.80	0.88	2.61	0.97	1.94

Table 3. Non-conformal baseline heuristics (for classification tasks only). Top-k takes a target size ( $k$ ), and yields statically sized outputs. Naive takes a target accuracy of  $1 - \epsilon$ , and yields dynamically sized outputs according to softmax probability mass.

**Task validity.** As per Theorem 4.3, we observe that our meta CP approach is valid, as the average accuracy always matches or exceeds the target performance level. Typically, meta CP is close to the target  $1 - \epsilon$  level for all  $\epsilon$ , which indicates that it is not overly conservative at any point (which improves the predictive efficiency). On the other hand, our full CP baseline is only close to the target accuracy when  $1 - \epsilon$  is near a multiple of  $\frac{1}{k+1}$ . This is visible from its “staircase”-like accuracy plot in Figure 4. We see that our sample-conditional approach is slightly conservative, as its accuracy typically exceeds  $1 - \epsilon$ . This is more pronounced for domains with smaller amounts of auxiliary data.

**Conditional coverage.** Figure 5 shows the accuracy of our meta quantile predictor  $\hat{P}_{1-\epsilon}$  as a function of  $k$ . As expected, as  $k$  grows,  $\hat{P}_{1-\epsilon}$  becomes more accurate. This lessens the need for large correction factors  $\Lambda(1 - \epsilon, \mathcal{I}_{\text{cal}})$ , and leads to task-conditional coverage, per Proposition 4.5.

**Baseline comparisons.** Table 3 gives the results for our non-conformal heuristics, Top-k and Naive. We see that both approaches under-perform our CP method in terms of efficiency. Comparing to Table 2, we see that we achieve similar accuracy to Top-k with smaller average sets (while also being able to set  $\epsilon$ ). Similarly, Naive is uncalibrated and gives conservative results: for a target  $\epsilon$  we obtain tighter prediction sets with our meta CP approach.

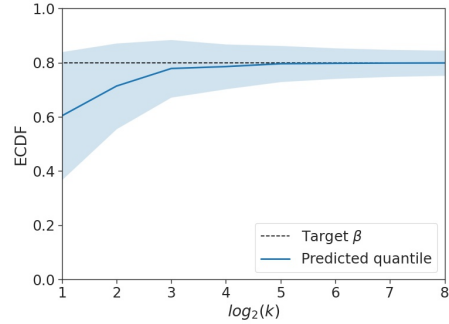


Figure 5. We measure the error in our quantile predictor  $\hat{S}_\beta$  (for  $\beta = 0.8$ ) on the CV task as a function of  $k$ . As  $k$  increases, the predictor begins to converge on an accurate  $\beta$ -quantile.

## 7 Conclusion

The ability to provide precise performance guarantees and make confidence-aware predictions is a critical element for many machine learning applications in the real world. Conformal prediction can afford remarkable finite-sample theoretical guarantees, but will suffer in practice when data is limited. In this paper, we introduced a novel and theoretically grounded approach to meta-learning few-shot conformal predictor using exchangeable collections of auxiliary tasks. Our results show that our method consistently improves performance across multiple diverse domains, and allow us to obtain meaningful and confident conformal predictors when using only a few in-task examples.

## Acknowledgements

We thank Kyle Swanson, the MIT NLP group, and anonymous reviewers for valuable feedback. AF is supported in part by the NSF GRFP. TS is supported in part by DSO grant DSOCL18002. This work is also supported in part by MLPDS and the DARPA AMD project.

## References

- Amodei, D., Olah, C., Steinhardt, J., Christiano, P., Schulman, J., and Mané, D. Concrete problems in ai safety. *arXiv preprint arXiv:1606.06565*, 2016.
- Angelopoulos, A. N., Bates, S., Malik, J., and Jordan, M. I. Uncertainty sets for image classifiers using conformal prediction. In *International Conference on Learning Representations (ICLR)*, 2021.
- Ashukha, A., Lyzhov, A., Molchanov, D., and Vetrov, D. Pitfalls of in-domain uncertainty estimation and ensembling in deep learning. In *International Conference on Learning Representations (ICLR)*, 2020.
- Bao, Y., Wu, M., Chang, S., and Barzilay, R. Few-shot text classification with distributional signatures. In *International Conference on Learning Representations (ICLR)*, 2020.
- Bates, S., Angelopoulos, A. N., Lei, L., Malik, J., and Jordan, M. I. Distribution free, risk controlling prediction sets. *arXiv preprint: arXiv 2101.02703*, 2020.
- Bertinetto, L., Henriques, J. F., Torr, P., and Vedaldi, A. Meta-learning with differentiable closed-form solvers. In *International Conference on Learning Representations (ICLR)*, 2019.
- Bottou, L. and Bousquet, O. The tradeoffs of large scale learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2008.
- Carlsson, L., Ahlberg, E., Boström, H., Johansson, U., and Linusson, H. Modifications to p-values of conformal predictors. In *Statistical Learning and Data Sciences*, 2015.
- Cauchois, M., Gupta, S., and Duchi, J. Knowing what you know: valid confidence sets in multiclass and multilabel prediction. *arXiv preprint: arXiv 2004.10181*, 2020.
- Chernozhukov, V., Wuthrich, K., and Zhu, Y. Distributional conformal prediction. *arXiv preprint: arXiv 1909.07889*, 2019.
- Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.
- Edwards, H. and Storkey, A. Towards a neural statistician. In *International Conference on Learning Representations (ICLR)*, 2017.
- Finn, C., Abbeel, P., and Levine, S. Model-agnostic meta-learning for fast adaptation of deep networks. In *International Conference on Machine Learning (ICML)*, 2017.
- Fisch, A., Schuster, T., Jaakkola, T., and Barzilay, R. Efficient conformal prediction via cascaded inference with expanded admission. In *International Conference on Learning Representations (ICLR)*, 2021.
- Gal, Y. and Ghahramani, Z. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *International Conference on Machine Learning (ICML)*, 2016.
- Graves, A. Practical variational inference for neural networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2011.
- Guo, C., Pleiss, G., Sun, Y., and Weinberger, K. Q. On calibration of modern neural networks. In *International Conference on Machine Learning (ICML)*, 2017.
- Han, X., Zhu, H., Yu, P., Wang, Z., Yao, Y., Liu, Z., and Sun, M. FewRel: A large-scale supervised few-shot relation classification dataset with state-of-the-art evaluation. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2018.
- Hernández-Lobato, J. M. and Adams, R. P. Probabilistic backpropagation for scalable learning of bayesian neural networks. In *International Conference on Machine Learning (ICML)*, 2015.
- Hirschfeld, L., Swanson, K., Yang, K., Barzilay, R., and Coley, C. W. Uncertainty quantification using neural networks for molecular property prediction. *arXiv preprint: arXiv 2005.10036*, 2020.
- Jiang, H., Kim, B., Guan, M., and Gupta, M. To trust or not to trust a classifier. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 5541–5552. 2018.
- Jiang, X., Osl, M., Kim, J., and Ohno-Machado, L. Calibrating predictive model estimates to support personalized medicine. *Journal of the American Medical Informatics Association*, 19(2):263–274, Mar-Apr 2012.
- Johansson, U., Ahlberg, E., Boström, H., Carlsson, L., Linusson, H., and Sönströd, C. Handling small calibration sets in mondrian inductive conformal regressors. In *Statistical Learning and Data Sciences*, 2015.
- Kivaranovic, D., Johnson, K. D., and Leeb, H. Adaptive, distribution-free prediction intervals for deep networks. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2020.
- Lake, B. M., Salakhutdinov, R., and Tenenbaum, J. B. Human-level concept learning through probabilistic program induction. *Science*, 350(6266):1332–1338, 2015. ISSN 0036-8075.

- Lakshminarayanan, B., Pritzel, A., and Blundell, C. Simple and scalable predictive uncertainty estimation using deep ensembles. In *Advances in Neural Information Processing Systems (NeurIPS)*. 2017.
- Lee, K., Lee, H., Lee, K., and Shin, J. Training confidence-calibrated classifiers for detecting out-of-distribution samples. In *International Conference on Learning Representations (ICLR)*, 2018.
- Lei, J., G’Sell, M., Rinaldo, A., Tibshirani, R. J., and Wasserman, L. Distribution-free predictive inference for regression. *Journal of the American Statistical Association*, 113(523):1094–1111, 2018.
- Linusson, H., Johansson, U., Boström, H., and Löfström, T. Efficiency comparison of unstable transductive and inductive conformal classifiers. In *Artificial Intelligence Applications and Innovations*, 2014.
- Mayr, A., Klambauer, G., Unterthiner, T., Steijaert, M., Wegner, J., Ceulemans, H., Clevert, D.-A., and Hochreiter, S. Large-scale comparison of machine learning methods for drug target prediction on chembl. *Chemical Science*, 9, 06 2018. doi: 10.1039/C8SC00148K.
- Neal, R. M. *Bayesian Learning for Neural Networks*. Springer-Verlag, 1996. ISBN 0387947248.
- Niculescu-Mizil, A. and Caruana, R. Predicting good probabilities with supervised learning. In *International Conference on Machine Learning (ICML)*, 2005.
- Nouretdinov, I., Melluish, T., and Vovk, V. Ridge regression confidence machine. In *International Conference on Machine Learning (ICML)*, 2001.
- Papadopoulos, H. Inductive conformal prediction: Theory and application to neural networks. In *Tools in Artificial Intelligence*, chapter 18. IntechOpen, Rijeka, 2008.
- Pennington, J., Socher, R., and Manning, C. GloVe: Global vectors for word representation. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2014.
- Romano, Y., Patterson, E., and Candes, E. Conformalized quantile regression. In *Advances in Neural Information Processing Systems (NeurIPS)*. 2019.
- Romano, Y., Barber, R. F., Sabatti, C., and Candès, E. With malice toward none: Assessing uncertainty via equalized coverage. *Harvard Data Science Review*, 4 2020.
- Shafer, G. and Vovk, V. A tutorial on conformal prediction. *Journal of Machine Learning Research (JMLR)*, 9: 371–421, June 2008.
- Snell, J., Swersky, K., and Zemel, R. Prototypical networks for few-shot learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.
- Tibshirani, R. J., Foygel Barber, R., Candes, E., and Ramdas, A. Conformal prediction under covariate shift. In *Advances in Neural Information Processing Systems (NeurIPS)*. 2019.
- Vinyals, O., Blundell, C., Lillicrap, T., kavukcuoglu, k., and Wierstra, D. Matching networks for one shot learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2016.
- Vovk, V. Conditional validity of inductive conformal predictors. In *Proceedings of the Asian Conference on Machine Learning*, 2012.
- Vovk, V., Gammerman, A., and Shafer, G. *Algorithmic Learning in a Random World*. Springer-Verlag, Berlin, Heidelberg, 2005.
- Wang, Y., Yao, Q., Kwok, J. T., and Ni, L. M. Generalizing from a few examples: A survey on few-shot learning. volume 53. Association for Computing Machinery, 2020.
- Yang, K., Swanson, K., Jin, W., Coley, C., Eiden, P., Gao, H., Guzman-Perez, A., Hopper, T., Kelley, B., Mathea, M., Palmer, A., Settels, V., Jaakkola, T., Jensen, K., and Barzilay, R. Analyzing learned molecular representations for property prediction. *Journal of Chemical Information and Modeling*, 59(8):3370–3388, 2019.
- Zaheer, M., Kottur, S., Ravanbakhsh, S., Poczos, B., Salakhutdinov, R. R., and Smola, A. J. Deep sets. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.