Journal of Machine Learning Research 25 (2024) 1-45 Submitted 3/23; Revised 5/24; Published 5/24

Predictive Inference with Weak Supervision

Maxime Cauchois MAXIME.CAUCHOIS@GMAIL.COM
Department of Statistics
Stanford University
Stanford, CA 94305-4020, USA

Suyash Gupta SUYASH028@GMAIL.COM
Department of Statistics
Stanford University
Stanford, CA 94305-4020, USA

Alnur Ali ALNURALI@GMAIL.COM
Department of Statistics
Stanford University
Stanford, CA 94305-4020, USA

John Duchi JDUCHI@STANFORD.EDU
Department of Statistics and electrical Engineering
Stanford University
Stanford, CA 94305-4020, USA

Editor: Daniel Hsu

Abstract

The expense of acquiring labels in large-scale statistical machine learning makes partially and weakly-labeled data attractive, though it is not always apparent how to leverage such data for model fitting or validation. We present a methodology to bridge the gap between partial supervision and validation, developing a conformal prediction framework to provide valid predictive confidence sets—sets that cover a true label with a prescribed probability, independent of the underlying distribution—using weakly labeled data. To do so, we introduce a (necessary) new notion of coverage and predictive validity, then develop several application scenarios, providing efficient algorithms for classification and several large-scale structured prediction problems. We corroborate the hypothesis that the new coverage definition allows for tighter and more informative (but valid) confidence sets through several experiments.

Keywords: Conformal inference, Confidence sets, Coverage validity, Weak supervision, Partial labels

1. Introduction

Consider the typical supervised learning pipeline that we teach students learning statistical machine learning: we collect data in (X, Y) pairs, where Y is a label or target to be predicted; we pick a model and loss measuring the fidelity of the model to observed data; we choose the model minimizing the loss and validate it on held-out data. This picture obscures what is becoming one of the major challenges in this endeavor: that of actually collecting high-quality labeled data (Sculley et al., 2015; Donoho, 2017; Ratner et al., 2017).

Hand labeling large-scale training sets is often impractically expensive. Consider, as simple motivation, a ranking problem: a prediction is an ordered list of a set of items, yet available feedback is likely to be incomplete and partial, such as a top element (for example, in web search a user clicks on a single preferred link, or in a grocery, an individual buys one kind of milk but provides no feedback on the other brands present). Developing methods to leverage such partial and weak feedback is therefore becoming a major focus, and researchers have developed methods to transform weak and noisy labels into a dataset with strong, "gold-standard" labels (Ratner et al., 2017; Zhang et al., 2017).

In this paper, we adopt this weakly labeled setting, but instead of considering model fitting and the construction of strong labels, we focus on validation, model confidence, and predictive inference, moving beyond point predictions and single labels. Our goal is to develop methods to rigorously quantify the confidence a practitioner should have in a model given only weak labels. First consider the standard supervised learning scenario for data $(X, Y) \in \mathcal{X} \times \mathcal{Y}$: here, given a desired confidence level α, the goal, rather than to provide point estimates $\widehat{Y}$ of Y given X, is to give a confidence set mapping $\widehat{C}_n$ based on $(X_i, Y_i)_{i=1}^n$ that guarantees the distribution-free coverage

$$\mathbb{P}\left[Y_{n+1} \in \widehat{C}_n(X_{n+1})\right] \geq 1 - \alpha, \tag{1}$$

where (X_{n+1}, Y_{n+1}) is a new observation following the same distribution as the first n points. Conformal inference provides precisely these guarantees (Vovk et al., 2005; Lei, 2014; Lei and Wasserman, 2014; Lei et al., 2018; Barber et al., 2021).

There are many scenarios, however, where it is natural to transition away from this strongly supervised setting with fully labeled examples. Above we note ranking: individuals are very unlikely to provide full feedback (Ailon et al., 2008; Duchi et al., 2013; Negahban et al., 2016). In multi-label image classification (Boutell et al., 2004; Elisseeff and Weston, 2001), a labeler may identify a few items in a given scene but not all, leading to partial labeled feedback. A major challenge in industrial machine learning deployment is to monitor models once they are in production, where it may be challenging to collect high-quality labels, but weak supervision—in the form of clicks on a recommended website, or agreeing to a suggested text message completion—is relatively easy and cheap to collect. In all of these, developing valid confidence sets and measures for our predictions is of growing importance, as we wish for models to be trustable, usable, verifiable.

With this as motivation, we consider supervised learning problems where, instead of directly observing the ground truth labels $\{Y_i\}_{i=1}^n$, we observe only noisy partial labeling. We make this formal in two equivalent ways. In the first, for each instance $i \in [n]$, there exists a (random) function $\varphi_i : \mathcal{Y} \to \mathcal{Y}^{\text{weak}}$, that belongs to a set $\Phi \subset \{\mathcal{Y} \to \mathcal{Y}^{\text{weak}}\}$ of partially supervising functions, such that we only observe $Y_i^{\text{weak}} = \varphi_i(Y_i)$. Equivalently, the pair $(Y_i^{\text{weak}}, \varphi_i)$ specifies a weak set $W_i \subset \mathcal{Y}$ that contains Y_i:

$$W_i := \left\{y \in \mathcal{Y} \mid \varphi_i(y) = Y_i^{\text{weak}}\right\} \subset \mathcal{Y}, \tag{2}$$

so that we observe a set W_i consistent with Y_i. Instead of observing strong labels $(X_i, Y_i) \stackrel{\text{iid}}{\sim} P$, we thus observe only $(X_i, \varphi_i, Y_i^{\text{weak}})_{i=1}^n$. Consider for instance a supervised problem where the goal is to rank elements of the set $\mathcal{C} := \{C_1, C_2, C_3, C_4\}$, hence each label

$Y \in \mathfrak{S}(\mathcal{C}, [4])$. In this setting, assume that the j-th data point has a strong label Y_j such that $Y_j(C_1) > Y_j(C_2) > Y_j(C_3) > Y_j(C_4)$, but we only observe the partial ranking $Y_j(C_1) > Y_j(C_3)$. This translates in our framework to having φ_i be a mapping from the set of permutations of $K = 4$ elements to the set of all partial rankings (equivalently the collection of sets of binary comparisons), and φ_j be the mapping $y \in \mathfrak{S}(\mathcal{C}, [4]) \to (1\{y(C_1) > y(C_3\})$. Other individuals may provide more or less information about their own personal rankings, but the key assumption of the paper is to assume that these random preferences are exchangeable with the future test point: if 50% data points rank C_1 and C_3 in the present dataset, there is a 50% chance that the test user also cares about it.

We emphasize that in this paper, φ_i is a random preference function with a crucial dual function. On the one hand, it describes which fraction of the ground truth label we actually observe in the partially labeled dataset. On the other hand, it describes which piece of information from the ground truth label actually matters to the test user, in short what they are likely to "query". The key assumption of this paper is to assume that these two distributions (partial feedback acquisition and future user query) coincide, hence that providing a label $y \in \mathcal{Y}$ that correctly answers the query (such that $\varphi_{n+1}(y) = Y_{n+1}^{\text{weak}}$) is beneficial to the test user. We argue that the assumption is plausible in various settings, for instance in item ranking problems, where an individual is intuitively more likely to both provide feedback and care about the elements at the top of their ranking. Finally, without loss of generality, one can always assume that $\mathcal{Y}^{\text{weak}} = 2^{\mathcal{Y}}$, since any preference function implicitly maps each element $y \in \mathcal{Y}$ to a subset of $\mathcal{Y}$ (containing of course y, see equation (2)).

We consider two fundamental questions in this weakly-labeled setting: (i) for what is it even possible to provide (distribution-free) coverage (e.g., true labels Y, weak labels W, or something else), and (ii) what methods can guarantee coverage?

While a first goal would be to produce a confidence mapping using $(X_i, Y_i^{\text{weak}}, \varphi_i)_{i=1}^n$ guaranteeing the coverage (1), as we prove in Section 2.1, this would in general produce large and hence, uninformative confidence sets. We therefore relax our coverage desiderata, instead seeking a confidence set $\widehat{C}_n : \mathcal{X} \rightrightarrows \mathcal{Y}$ that covers the weak counterpart $Y_{n+1}^{\text{weak}} = \varphi_{n+1}(Y_{n+1})$ of the true label in the sense that

$$\mathbb{P}\left[\widehat{C}_n(X_{n+1}) \cap W_{n+1} \neq \emptyset\right] = \mathbb{P}\left[\exists y \in \widehat{C}_n(X_{n+1}) \text{ s.t. } \varphi_{n+1}(y) = Y_{n+1}^{\text{weak}}\right] \geq 1 - \alpha. \quad (3)$$

One could certainly think of using standard conformal inference approaches (Vovk et al., 2005; Lei, 2014; Barber et al., 2021) to obtain this, but the direct application of conventional conformal inference methods to make predictions in the space of weak labels is practically impossible since the space of weak labels is a priori unknown. We do not know what the user's preference function φ will be. For example, in situations like item rankings, it is uncertain which item attributes are most important to a given user in the test sample. Hence, we wish to return usable labels and configurations in the actual target space $\mathcal{Y}$ of interest. The condition (3) is weaker than the initial coverage (1), and any confidence set satisfying the former will also satisfy the latter, though it allows smaller confidence set sizes. A major challenge is that the function φ_{n+1} representing an individual's weak supervision is *a priori* unknown (e.g., we do not know precisely what items a labeler will label in an image ahead of time). Indeed, if we observe φ_{n+1} prior to our prediction, a simple and optimal way

to achieve coverage (3) follows. First, construct a valid confidence set mapping $\widehat{C}_{n,\text{weak}} : \mathcal{X} \rightrightarrows \mathcal{Y}^{\text{weak}}$ for Y_{n+1}^{weak} using classical conformal methodology Vovk et al. (2005); Lei (2014); Barber et al. (2021), which as in (1) would guarantee $\mathbb{P}(Y_{n+1}^{\text{weak}} \in \widehat{C}_{n,\text{weak}}(X_{n+1})) \geq 1 - \alpha$, Then define $\widehat{C}_n(x) \subset \mathcal{Y}$ to include a single $y \in \mathcal{Y}$ for each $y^{\text{weak}} \in \widehat{C}_{n,\text{weak}}(x)$ such that $\varphi_{n+1}(y) = y^{\text{weak}}$. Such an approach is unfortunately impossible: we do not know ahead of time if an individual cares only about the top item of her ranking or requires a ranking accurate up to the 10th item.

Given the subtleties of coverage (3), we dedicate Section 2 to question (i) above: what types of coverage are even possible. We devote Sections 3 and 4 to question (ii): the development of methodologies that can guarantee the coverage (3). In the former (Sec. 3), we provide a general recipe, while in Section 4 we provide more tailored methods for large output spaces, such as those in structured prediction. To provide some initial insights into the methods and potential applications, we provide experiments on several real-world domains in Section 5.

1.1 Related Work

An extensive line of work addresses prediction with partially labeled data. The major focus is on strong label recovery under weak supervision, including in multiclass (Cour et al., 2011; Nguyen and Caruana, 2008) and multilabel Yu et al. (2014) tasks as well as structured prediction problems, such as ranking (Hüllermeier et al., 2008; Korba et al., 2018), segmentation (Triggs and Verbeek, 2008; Papandreou et al., 2015), and natural language processing (Fernandes and Brefeld, 2011; Mayhew et al., 2019). More recent work tackles constructing strongly labeled datasets from disparate weak supervision tasks (Ratner et al., 2017; Zhang et al., 2017), while the papers (Cid-Sueiro et al., 2014; van Rooyen and Williamson, 2018; Cabannes et al., 2020) provide generic theoretical conditions allowing strong label recovery. Yet this literature focuses primarily on point prediction problems, where a model only returns a single label with the (putative) highest likelihood, in contrast to our confidence-based approach, which provides calibrated uncertainty estimates and guarantees valid confidence sets with virtually no distributional assumptions.

Our work also connects to the substantial literature on conformal inference, where the goal is to provide valid predictive confidence sets (1). Vovk et al. (2005) introduce the main techniques—that examples are exchangeable, and so essentially can provide p-values for significance of one-another—and suggest the simple and generic split-conformal algorithm for building valid confidence sets. Essentially all conformalized confidence sets offer the coverage guarantee (1), so it is of interest to improve various aspects of the mappings $\widehat{C}_n$. For example, works focus on improving the precision of these methods and optimizing average confidence set size (Lei et al., 2018; Sadinle et al., 2019; Hechtlinger et al., 2019; Romano et al., 2019a; Angelopoulos et al., 2020; Barber et al., 2021), or on bridging the gap with other forms of coverage, like classwise (Sadinle et al., 2019) or conditional (Romano et al., 2019b; Barber et al., 2021; Cauchois et al., 2021; Romano et al., 2020; Cauchois et al., 2020; Gupta and Rothenhäusler, 2021; Duchi et al., 2024; Gupta, 2022; Cauchois et al., 2022) coverage.

Along these lines, Bates et al. (2021) generalize conformal inference to offer error control with respect to loss functions beyond the 0-1 loss (coverage or non-coverage) central to the

guarantee (1), taking structured prediction problems as motivation—as we do. Bates et al. (2021) focus on settings where the loss function naturally reflects the structure of the label space $\mathcal{Y}$, such as hierarchical classification problems where one wishes to label an example X at a resolution (level of the tree) appropriate to the confidence with which it can be labeled. We view our approaches as complementary to theirs: their approaches make sense for scenarios with fully labeled data in which a particular loss function is natural, for example in tree-structured hierarchical classification, where a prediction can be made at a given level in the tree. Conversely, approaches are sensible when one receives weakly supervised data and wishes to make a single good prediction; think of a grocery store deciding which of a large collection of shaving creams to stock, a ranking problem with the where one wishes to make sure that each individual's desired shaving cream is stocked (in this context, this is the guarantee (3)). In that respect, our approach relates to the expanded admission problem Fisch et al. (2021), which allows for multiple labels to be "admissible", except that we do not observe to strongly supervised labels in our setting. Consequently, we motivate our distinct coverage guarantees from a set of impossibility results we present in the next section. Additionally, we pay special attention (see Sec. 4) to developing practical algorithms that scale to large label spaces, an important consideration with real-world weak supervision.

Notation Throughout this paper, $[n]$ stands for the set $\{1, 2, \ldots, n\}$. We use $C : \mathcal{X} \rightrightarrows \mathcal{Y}$ to denote a set valued mapping $C : \mathcal{X} \to 2^{\mathcal{Y}} := \{W \mid W \subset \mathcal{Y}\}$. P is either the probability distribution generating the data $(X, Y, \varphi) \in \mathcal{X} \times \mathcal{Y} \times \Phi$, or equivalently $(X, Y, W) \in \mathcal{X} \times \mathcal{Y} \times 2^{\mathcal{Y}}$, as both notations are equivalent for our purpose, and $U \sim \mathsf{Uni}[0, 1]$ defines a uniform random variable on $[0, 1]$. $\mathfrak{S}(U, V)$ is the set of bijections between two sets U and V, using the shorthand $\mathfrak{S}_K := \mathfrak{S}([K], [K])$; (i, j) is the transposition of elements i and $j \in [K]$, and for $k \in \mathbb{N}$, $\Delta_k := \{p \in \mathbb{R}_+^k \mid p^T \mathbf{1} = 1\}$ is the space of probability distributions on $[k]$.

2. Conformal inference with weakly supervised data

The starting point of this paper is to delineate realistic goals in weak-conformal inference by determining what is actually possible—as we show, a form of weak coverage—and what is unachievable. To that end, we demonstrate that strong coverage (1), while desirable, this may yield prediction sets that are potentially expansive and challenging to interpret. We thus relax our goals, presenting a general weak conformal scheme (Section 2.2) that relies on weakly supervised data. For example, in ranking four items $\mathcal{C} := \{C_1, C_2, C_3, C_4\}$, assume that the j-th data point has a weak label Y_j such that $Y_j(C_1) > Y_j(C_3)$, as long as the prediction set has element that preserves the ordering of C_1 and C_3, it serves the purpose. Aiming for strong coverage leads to a prediction set of size at least $4!/2! = 12$, making interpretation challenging.

2.1 The strong coverage dilemma with partially supervised data

Consider a fully supervised classification setting with feature space $\mathcal{X}$ and output space $\mathcal{Y}$, and let P_{strong} be a joint distribution on $\mathcal{X} \times \mathcal{Y}$ representing strong, as opposed to weak, supervision. In this fully supervised setting, we observe samples $(X_i, Y_i) \overset{\text{iid}}{\sim} P_{\text{strong}}$,

in contrast to observing the weak label set $W \subset \mathcal{Y}$ satisfying only $Y \in W$. We first require definitions of consistency and validity.

Definition 1 *A probability distribution P on $(X, Y, W) \in \mathcal{X} \times \mathcal{Y} \times 2^{\mathcal{Y}}$ is* consistent *if $P(Y \in W) = 1$. For any consistent distribution P, P_{weak} and P_{strong} denote the marginal distributions of (X, W) and (X, Y), respectively, when $(X, Y, W) \sim P$.*

Definition 2 *Let $\widehat{C}_n : \mathcal{X} \rightrightarrows \mathcal{Y}$ be a (potentially randomized) procedure depending only on the weakly supervised sample $(X_i, W_i)_{i=1}^n \in \mathcal{X} \times 2^{\mathcal{Y}}$. Then $\widehat{C}_n$ provides* $(1-\alpha)$-strong distribution free coverage *if for all consistent distributions P on $\mathcal{X} \times \mathcal{Y} \times 2^{\mathcal{Y}}$ and $(X_i, Y_i, W_i)_{i=1}^{n+1} \stackrel{\text{iid}}{\sim} P$, we have Eq. (1), i.e.*

$$\mathbb{P}\left[Y_{n+1} \in \widehat{C}_n(X_{n+1})\right] \geq 1 - \alpha,$$

and that $\widehat{C}_n$ provides $(1-\alpha)$-weak distribution free coverage *if it satisfies Eq. (3), i.e.*

$$\mathbb{P}\left[W_{n+1} \cap \widehat{C}_n(X_{n+1}) \neq \emptyset\right] \geq 1 - \alpha.$$

With these definitions, we can provide the (negative) result that, on average over the data set, any procedure satisfying strong distribution free coverage (1) must include every individual label $y \in W_{n+1}$ with probability at least $1-\alpha$. To formalize this, for a confidence set mapping $\widehat{C}_n : \mathcal{X} \rightrightarrows \mathcal{Y}$ constructed with $(X_i, W_i)_{i=1}^n$, define the function

$$p_n(x, y) \coloneqq \mathbb{P}\left(y \in \widehat{C}_n(x)\right),$$

which is the probability, taken over the weakly supervised sample $(X_i, W_i)_{i=1}^n$, that $\widehat{C}_n(x)$ contains the potential label y. We prove the following theorem in Appendix B.1.1.

Theorem 3 *Suppose that $\widehat{C}_n : \mathcal{X} \rightrightarrows \mathcal{Y}$ provides $(1-\alpha)$-strong distribution free coverage. Then for all consistent distributions P on $\mathcal{X} \times \mathcal{Y} \times 2^{\mathcal{Y}}$,*

$$\mathbb{E}_{\{(X_i,W_i)\}_{i=1}^{n+1} \stackrel{\text{iid}}{\sim} P_{\text{weak}}}\left[\inf_{y \in W_{n+1}} p_n(X_{n+1}, y)\right] \geq 1 - \alpha.$$

Theorem 3 essentially states that $\widehat{C}_n$ simultaneously includes each element $y \in W_{n+1}$ with probability at least $1-\alpha$. The theorem is generally not improvable, as W_{n+1} need not be a subset of $\widehat{C}_n(X_{n+1})$. Indeed, think of the trivial procedure $\widehat{C}_n$ that includes every label $y \in \mathcal{Y}$ independently with probability $1-\alpha$: it obviously satisfies strong distribution-free coverage but has no connection with W_{n+1}. As an additional immediate corollary, if the sets W contain at least a fixed number of labels, then so does $\widehat{C}_n(X_{n+1})$.

Corollary 4 *Suppose that $\widehat{C}_n : \mathcal{X} \rightrightarrows \mathcal{Y}$ provides $(1-\alpha)$-strong distribution free coverage, and that $P(|W| \geq L) = 1$ for some $L \geq 1$. Then*

$$\mathbb{E}_{\{(X_i,W_i)\}_{i=1}^{n+1} \stackrel{\text{iid}}{\sim} P_{\text{weak}}}\left[\left|\widehat{C}_n(X_{n+1})\right|\right] \geq L(1-\alpha).$$

Proof By Theorem 3,

$$\mathbb{E}\left[\left|\widehat{C}_n(X_{n+1})\right|\right] = \mathbb{E}\left[\sum_{y\in\mathcal{Y}} p_n(X_{n+1}, y)\right] \geq \mathbb{E}\left[|W_{n+1}| \inf_{y\in W_{n+1}} p_n(X_{n+1}, y)\right] \geq L(1-\alpha)$$

as claimed. ■

An alternative perspective is to consider large-sample limits; often, the procedure $\widehat{C}_n$ converges to some population confidence set mapping $C : \mathcal{X} \rightrightarrows \mathcal{Y}$ as $n \to \infty$, in that

$$\mathbb{E}\left[\left|\widehat{C}_n(X) \triangle C(X)\right|\right] \to 0 \tag{4}$$

as $n \to \infty$, where the expectation is over both the construction of $\widehat{C}_n$ and X independent of $(X_i, W_i) \stackrel{\text{iid}}{\sim} P_{\text{weak}}$. Typically, the limiting C is a (nearly) deterministic function[1] of x; for example, the standard construction (e.g. Vovk et al., 2005; Lei, 2014; Barber et al., 2021) takes $C(x) = \{y \in \mathcal{Y} \mid s(x, y) \leq \tau\}$ for some scoring function $s : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ and threshold τ, which is deterministic. In this case, we can show that we nearly have $W \subset C(X)$, so $C(X)$ must be large whenever W is. To formalize, let

$$\text{Det}_C(x) := \{y \in \mathcal{Y} \mid \mathbb{P}(y \in C(x)) \in \{0, 1\}\}$$

be the labels that are deterministically in *or* out of $C(x)$ (where the probability is over any randomization in the mapping C) so that $\text{Det}_C(x) = \mathcal{Y}$ whenever C is deterministic. Then can show that $W \subset C(X)$ with probability at least $1 - \alpha$:

Corollary 5 *Suppose that $\widehat{C}_n : \mathcal{X} \rightrightarrows \mathcal{Y}$ provides $(1-\alpha)$-strong distribution free coverage and satisfies the limit* (4)*. Then*

$$\mathbb{P}(W \cap \text{Det}_C(X) \subset C(X)) \geq 1 - \alpha.$$

See Appendix B.1.2 for a proof.

Theorem 3 and its corollaries suggest that any procedure achieving strong (distribution-free) coverage necessarily produces inefficient (large) confidence sets when one uses only weakly supervised data. Even in cases where there is implicitly a single correct label, such as the structured prediction problems Cabannes et al. (2020) consider, where the weak labels w that a single x supports (those for which $\mathbb{P}(W = w \mid X = x) > 0$) have a single label y in their intersection $\cap_{w:\mathbb{P}(w|x)>0}\{w\} = \{y\}$, never disallow large weak sets W. We thus must take a different tack, targeting new coverage desiderata.

An aside: regression. Although we primarily focus on classification (where $\mathcal{Y}$ is finite), our development applies equally to regression or other problems with continuous or infinite response sets, e.g., $\mathcal{Y} = \mathbb{R}$, as nothing in Theorems 3 or 7 requires $\mathcal{Y}$ to be any particular space. We leverage this in our experiments (Sec. 5) in the sequel to give numerical examples, touching on the $\mathbb{R}$-valued case here to demonstrate the analogues of our theoretical results.

1. In some cases, we use randomization over a single label to guarantee that $\mathbb{P}(Y \in C(X)) = 1 - \alpha$

As an example, weak sets W in the continuous case may be intervals, arising, for example, from measurements with limited resolution. We adapt Corollary 4 to regression by replacing counting measure with the Lebesgue measure Leb, where the response set $\mathcal{Y} = \mathbb{R}$ and the weak sets $W \subset \mathbb{R}$. Assuming that the weak sets all have a minimal volume, any valid confidence set mapping necessarily is large (on average) as well:

Corollary 6 *Suppose that $\widehat{C}_n : \mathcal{X} \rightrightarrows \mathcal{Y} = \mathbb{R}$ provides $(1-\alpha)$-strong distribution free coverage, and let $L > 0$. If $P(\mathrm{Leb}(W_i) \geq L) = 1$, for $i = 1, \ldots, n+1$, then*

$$\mathbb{E}\Big[\mathrm{Leb}\big(\widehat{C}_n(X_{n+1})\big)\Big] \geq L(1-\alpha).$$

An analogy to Corollary 5 follows nearly immediately from Corollary 6, just as Corollary 5 follows Corollary 4. That is, if there exists a deterministic set-valued mapping $C : \mathcal{X} \rightrightarrows \mathcal{Y}$ for which $\mathbb{E}[\mathrm{Leb}(\widehat{C}_n(X) \triangle C(X))] \to 0$, then $\mathbb{P}(W \subset \widehat{C}_n(X_{n+1})) \geq 1 - \alpha$.

2.2 A general weak-conformal scheme via scoring functions

The theoretical limitations we identify motivate the weak coverage (3) we target instead of the strong converage (1). Following our discussion above, the new coverage definition stems from two desiderata: if the problem is actually low-noise and there already exists a highly predictive model we can leverage to build our confidence sets—roughly, that conditional on x, a single label y belongs to the weak sets W with high probability and a model exists that can predict this y—then while we should return this singleton even if we cannot guarantee strong coverage. In the alternative perspective that we care only about the value $\varphi(Y)$—recall the weak set (2)—providing any y satisfying $\varphi(y) = Y^{\mathrm{weak}}$ should suffice for prediction. We turn now to provide our general weak conformalization scheme.

Our starting point is via the typical output of a machine-learned model, a scoring function $s : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ that ranks potential labels (or responses) y for an input example $x \in \mathcal{X}$. We treat $s(x, y)$ is a *non-conformity* score, meaning the model predicts that values of y for which $s(x, y)$ is small are more likely. Standard examples of such scoring functions include $s(x, y) \coloneqq -|y - \widehat{\mu}(x)|$ in regression, where $\widehat{\mu} : \mathcal{X} \to \mathbb{R}$ predicts $y \mid x$; or $s(x, y) \coloneqq -\log p_y(x)$ in multiclass classification, where $p_y(x)$ models the conditional probability of $y \mid x$. Throughout this section, we adopt a split-conformal perspective (Vovk et al., 2005; Barber et al., 2021), assuming the practitioner provides a scoring function independent of the sample $(X_i, \varphi_i, Y_i^{\mathrm{weak}})_{i=1}^n$ (the sample would typically be a *validation set*), and we show how to transform any such scoring function into a valid (weakly-covering) confidence mapping.

Algorithm 1 starts from a simple observation, assuming that the scoring function s is relatively good. Given a query function φ and weak label Y^{weak}—equivalently, the weak set $W = \{y \mid \varphi(y) = Y^{\mathrm{weak}}\}$—the most likely label should typically be the y satisfying $\varphi(y) = Y^{\mathrm{weak}}$ minimizing $s(x, y)$.
Note that a completely equivalent scheme to the scores (5) with label mappings φ and weak labels Y_i^{weak} uses weak sets W_i, where we replace the scores (5) with

$$S_i \coloneqq \min_{y \in W_i} s(X_i, y).$$

In either case, Algorithm 1 achieves valid coverage weak (3):

Algorithm 1 Partially supervised conformalization

Input: sample $\{(X_i, Y_i^{\text{weak}}, \varphi_i)\}_{i=1}^n$; score function $s : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ independent of the sample; desired coverage $1 - \alpha \in (0, 1)$
For each $i \in [n]$, compute

$$S_i \coloneqq \min_{y:\ \varphi_i(y) = Y_i^{\text{weak}}} s(X_i, y). \tag{5}$$

Set $\widehat{t}_n \coloneqq (1 + n^{-1})(1 - \alpha)$-quantile of $\{S_i\}_{i=1}^n$.
Return: predictive set mapping $\widehat{C}_n : \mathcal{X} \rightrightarrows \mathcal{Y}$ defined by

$$\widehat{C}_n(x) \coloneqq \left\{ y \in \mathcal{Y} \mid s(x, y) \le \widehat{t}_n \right\}.$$

Theorem 7 *Let $(X_i, Y_i, \varphi_i)_{i=1}^{n+1} \overset{\text{iid}}{\sim} P$ and $Y_i^{\text{weak}} = \varphi_i(Y_i)$ for $i \in [n+1]$. Then Algorithm 1 returns a confidence set mapping satisfying*

$$\mathbb{P}\left[\textit{There exists } y \in \widehat{C}_n(X_{n+1}) \textit{ s.t } \varphi_{n+1}(y) = \varphi_{n+1}(Y_{n+1}) = Y_{n+1}^{\text{weak}} \right] \ge 1 - \alpha.$$

Proof Let $S_i \coloneqq \min_{\varphi_i(y) = Y_i^{\text{weak}}} s(X_i, y)$ for each $i \in [n+1]$. By definition of $\widehat{C}_n$, we have

$$\left\{ y \in \widehat{C}_n(X_{n+1}) \mid \varphi_{n+1}(y) = Y_{n+1}^{\text{weak}} \right\} = \left\{ y \in \mathcal{Y} \mid \varphi_{n+1}(y) = Y_{n+1}^{\text{weak}} \text{ and } s(X_{n+1}, y) \le \widehat{t}_n \right\},$$

which is nonempty if and only if

$$S_{n+1} \coloneqq \min_{\varphi_{n+1}(y) = Y_{n+1}^{\text{weak}}} s(X_{n+1}, y) \le \widehat{t}_n.$$

As $\{S_i\}_{i=1}^{n+1}$ are i.i.d., this occurs with probability at least $1 - \alpha$ (e.g. Tibshirani et al., 2019, Lemma 1). ■

3. Constructing effective conformal prediction sets

Algorithm 1 provides a generic method for conformalization in the presence of partially supervised data, and it makes no assumptions on the input score function s. Though the coverage guarantee (3) holds regardless, we can delineate a few additional desiderata that the predictive sets and score functions s should satisfy to make them more practically useful, which is our focus in this section:

- The score function s must allow the practitioner to efficiently carry out the computation of the partial infimum scores (5).
- The lower level sets $\widehat{C}_n(x) = \{y \in \mathcal{Y} \mid s(x, y) \le \widehat{t}_n\}$ should be efficiently representable.
- The confidence sets $\widehat{C}_n(x)$ should be small, as smaller confidence sets (for a fixed confidence level α) carry more information.

Deferring our discussion of computational efficiency to Section 4, in this section we only focus on the last desideratum, implicitly assuming computation is tractable (e.g., if $\mathcal{Y}$ is small). We first (Sec. 3.1) develop conditions sufficient for optimally-sized confidence sets to even exist—a few subtleties arise—before giving greedy algorithms for confidence set-size minimization, describing their properties, and providing a few optimality guarantees in Sections 3.2 and A.

3.1 Size-optimal scoring mechanism

As in standard approaches to conformal inference (Lei, 2014; Lei and Wasserman, 2014; Barber et al., 2021), we aim to construct a confidence set mapping $\widehat{C}_n$ with minimal average size over $X \sim P_X$. Our starting point is simply to define size-optimality, where to achieve exact coverage and size guarantees, we allow randomization of our confidence sets via an independent variable $U \sim \mathsf{Uni}[0,1]$.

Definition 8 *A randomized confidence set mapping $C_{1-\alpha} : \mathcal{X} \times [0,1] \rightrightarrows \mathcal{Y}$ is* marginally size-optimal *at level α if it solves*

$$\begin{aligned} \underset{C:\mathcal{X}\times[0,1]\rightrightarrows\mathcal{Y}}{\text{minimize}} \; & \mathbb{E}_{X,U\sim\mathsf{Uni}[0,1]}\left[|C(X,U)|\right] \\ \text{subject to } & \mathbb{P}(W \cap C(X,U) \neq \emptyset) \geq 1-\alpha. \end{aligned} \tag{Marg}$$

It is conditionally size-optimal *at level α if for almost every $x \in \mathcal{X}$, $C(x,\cdot)$ solves*

$$\underset{C:[0,1]\rightrightarrows\mathcal{Y}}{\text{minimize}} \; \left\{\mathbb{E}_{U\sim\mathsf{Uni}[0,1]}\left[|C(U)|\right] \;\; s.t. \;\; \mathbb{P}(W \cap C(U) \neq \emptyset \mid X = x) \geq 1-\alpha\right\}. \tag{Cond}$$

Even with full knowledge of the distribution P, techniques for finding marginally size-optimal confidence sets (Marg) are not immediately apparent; as a consequence, we focus on the conditional case first. Even in this case, it is in general non-trivial to obtain smallest confidence sets. Yet as we follow the standard practice (Barber et al., 2021; Lei, 2014; Vovk et al., 2005) in conformal prediction of defining confidence sets via the scores s (recall Alg. 1) as $C_t(x) = \{y \mid s(x,y) \leq t\}$, our confidence sets have the natural nesting property that $C_t(x) \subset C_{t'}(x)$ whenever $t < t'$. Abstracting away the particular form of C to enable a purely set-based focus, we thus consider nested confidence sets, where we show that optimality guarantees are possible.

Definition 9 *A collection of mappings $\{C_\eta : \mathcal{X} \times [0,1] \rightrightarrows \mathcal{Y}\}_{\eta\in(0,1)}$ is* nested *if*

$$P(C_{\eta_1}(X,U) \subset C_{\eta_2}(X,U)) = 1 \text{ for all } 0 < \eta_1 < \eta_2 < 1.$$

There is an immediate equivalence between score-based conformalization schemes and nested collections of confidence mappings (Gupta et al., 2022): we simply define

$$s^{\text{nest}}(x,y,u) := \inf\left\{\eta \in (0,1) \mid y \in C_\eta(x,u)\right\}. \tag{6}$$

The next lemma formalizes this equivalence (see Appendix B.2.1 for a proof).

Lemma 10 *Assume the confidence set mappings* $\{C_\eta\}_{\eta\in(0,1)}$ *are nested and* $s^{\text{nest}}(x,y,U)$ *has continuous distribution for* $U \sim \mathsf{Uni}[0,1]$. *Then*

$$C_\eta(x,U) = \left\{y \in \mathcal{Y} \mid s^{\text{nest}}(x,y,U) \le \eta\right\} \quad \textit{with } U\textit{-probability } 1.$$

That is, obtaining weak coverage for nested confidence mappings is equivalent to obtaining weak coverage using the scoring function s^{nest}, which Alg. 1 provides, that is, equivalent to choosing the smallest $\eta \in (0,1)$ such that $\mathbb{P}(W \cap C_\eta(X,U) \neq \emptyset) \ge 1-\alpha$. A second useful distributional property of the nested scores (6) is that, assuming the confidence sets C_η are conditionally valid, we can provide strong distributional results on s^{nest}. To make this precise, we say that C_η is *conditionally valid for the weak labels* W if for each $\eta \in (0,1)$ and with $\mathbb{P}$-probability 1 over X,

$$\mathbb{P}\left(C_\eta(x,U) \cap W \neq \emptyset \mid X = x\right) = \eta. \tag{7}$$

We then have the following uniformity property as an immediate consequence of Lemma 10:

Lemma 11 *In addition to the conditions of Lemma 10, assume that* C_η *is conditionally valid* (7) *for the weak label* W. *Then the minimum score* (5) *is independent of* X *and satisfies*

$$\inf_{y\in W} s^{\text{nest}}(x,y,U) \sim \mathsf{Uni}[0,1].$$

Proof With U-probability 1, $\inf_{y\in W} s^{\text{nest}}(x,y,U) \le \eta$ if and only if $C_\eta(x,U)\cap W \neq \emptyset$, and so $\mathbb{P}(\inf_{y\in W} s^{\text{nest}}(x,y,U) \le \eta \mid X = x) = \mathbb{P}(W \cap C_\eta(x,U) \neq \emptyset \mid X = x) = \eta$. ■

A comparable result to Lemma 11 was previously asserted in Proposition 1 of Einbinder et al. (2022), where the affiliated conformity scores introduced therein also induce prediction sets that satisfy the nested property. To illustrate this lemma, suppose that there exist nested conditionally size-optimal mappings $\{C_\eta^{\text{cond}}\}_{\eta\in(0,1)}$ solving problem (Cond): in that case, they satisfy the conditions for application of Lemma 11 so that the induced scores S_i are uniform; Alg. 1 will thus compute $\widehat{t}_n = (1-\alpha) + O_P(n^{-1/2})$ as $\widehat{t}_n$ is the $(1-\alpha)$ quantile of $S_i \overset{\text{iid}}{\sim} \mathsf{Uni}[0,1]$. So—in the case that we have (near) conditional coverage—Alg. 1 maintains it. Notably, given a score function s, not necessarily the nested score (6), but strong in the sense that it models (X,Y) well enough that for each α, we can choose t so that $\mathbb{P}(s(x,Y) \le t \mid X = x) = \alpha$, then the confidence sets Algorithm 1 returns are indeed nested, and Lemma 11 applies to the induced nested score s^{nest}. Unfornately, optimal nested sets need not always exist (see Example 1 below), but we can provide natural conditions on the distribution of $W \mid X = x$ sufficient to allow such nested coverage, which we do in the next subsection.

3.1.1 From conditionally to marginally valid confidence sets

Our initial criterion (Marg) is purely marginal: we wish to compute a marginally size-optimal confidence set. Conveniently, conditionally size-optimal mappings can yield marginally size-optimal problems. In particular, assume that the mappings $\{C_\eta^{\text{cond}}\}_{\eta\in(0,1)}$ are conditionally size-optimal (Cond) and satisfy $\mathbb{P}(W \cap C^{\text{cond}}(x,U) \neq \emptyset \mid X = x) \ge \eta$. The

following proposition shows how to transform these into marginally size-optimal confidence sets.

Proposition 12 *Let the mappings* $\{C_\eta^{\text{cond}}\}$ *be conditionally size-optimal* (Cond) *as above, and define the average* $\text{size}(x, \eta) \coloneqq \mathbb{E}_U[|C_\eta^{\text{cond}}(x, U)|]$. *Let* s_{marg} *be any minimizer of*

$$\mathbb{E}\left[\text{size}(X, s(X))\right] \quad s.t. \quad \mathbb{E}[s(X)] \geq 1 - \alpha$$

over $s : \mathcal{X} \to [0, 1]$. *Then a solution to the initial marginal problem* (Marg) *is*

$$C_{1-\alpha}^{\text{marg}}(x, u) \coloneqq C_{s_{\text{marg}}(x)}^{\text{cond}}(x, u).$$

More directly, any conditionally size-optimal sets—which are at least easier to *characterize* as they need only randomize over $U \sim \mathsf{Uni}[0, 1]$—yield marginally size-optimal confidence sets in a relatively straightforward way: one chooses the probability of miscoverage, $s(x)$, minimizing the expected confidence set size.

Proof That $C_{1-\alpha}^{\text{Marg}}$ provides valid $1 - \alpha$ coverage is nearly immediate: by conditional size optimality, we have $\mathbb{P}(W \cap C_{1-\alpha}^{\text{marg}}(X, U) \neq \emptyset) = \mathbb{E}[t_{\text{marg}}(X)] \geq 1 - \alpha$.

Let C be any confidence set mapping such that $\mathbb{P}(W \cap C(X, U) \neq \emptyset) \geq 1 - \alpha$, and define $s_C(x) \coloneqq \mathbb{P}(W \cap C(x, U) \neq \emptyset \mid X = x) \in [0, 1]$, which satisfies $\mathbb{E}[s_C(X)] \geq 1 - \alpha$. By assumption on C^{cond}, for each fixed $x \in \mathcal{X}$, the set $C_{s_C(x)}^{\text{cond}}(x, U)$ is size-optimal (Cond) at level $s_C(x)$, so that for P_X-almost every $x \in \mathcal{X}$, we have

$$\text{size}(x, s_C(x)) = \mathbb{E}_{U \sim \mathsf{Uni}[0,1]}\left[|C_{s_C(x)}^{\text{cond}}(x, U))|\right] \leq \mathbb{E}_{U \sim \mathsf{Uni}[0,1]}\left[|C(x, U)|\right].$$

Integrating both sides of the inequality over $X \sim P_X$, and using the assumed optimality condition on s_{marg}, we obtain

$$\mathbb{E}\left[\text{size}(X, s_{\text{marg}}(X))\right] \leq \mathbb{E}\left[\text{size}(X, s_C(X))\right] \leq \mathbb{E}_{X,U}\left[|C(X, U))|\right].$$

The left-hand size is the average size of $C_{1-\alpha}^{\text{margin}}$. ■

3.2 Greedy algorithms for confidence set-size minimization

Given the distribution—or a model of the distribution—of the weak set W conditional on x, we propose a natural greedy algorithm to construct a confidence set satisfying the weak coverage constraint: at each step, Algorithm 2 adds the label that increases coverage the most until the confidence set achieves a desired level. Algorithm 2 draws inspiration from Algorithm 1 developed in Romano et al. (2020), where the authors formulated conformal inference methods tailored for categorical and unordered response labels. These methods not only ensure valid marginal coverage but also afford approximate conditional coverage. As we show presently, there are natural families of distributions where this greedy algorithm is optimal; however, there are failure modes, of which we also provide an example. In Appendix A, we relate this greedy construction to submodular optimization to provide general guarantees of confidence set size and coverage.

Algorithm 2 Greedy weakly supervised scoring mechanism

Input: model for the distribution of W given $X = x$; coverage rate $\eta \in (0, 1)$

for each $j \in [K]$ define recursively

$$y_j(x) \coloneqq \operatorname*{argmax}_{y \in \mathcal{Y}} P\left(y \in W, \bigcap_{i=1}^{j-1} \{y_i(x) \notin W\} \mid X = x\right).$$

for each $j \in [K]$ define $C^{\mathrm{gr},j}(x) \coloneqq \{y_i(x) \mid i \le j\}$ and **set**

$$j(x, \eta) \coloneqq \min\left\{j \in [K] \mid P\left(W \cap C^{\mathrm{gr},j}(x) \neq \emptyset \mid X = x\right) \ge \eta\right\}.$$

set

$$t_\eta(x) \coloneqq \frac{\eta - P(C^{\mathrm{gr},j(\eta,x)-1}(x, U) \cap W \neq \emptyset \mid X = x)}{P(C^{\mathrm{gr},j(x,\eta)}(x, U) \cap W \neq \emptyset \mid X = x) - P(C^{\mathrm{gr},j(x,\eta)-1}(x, U) \cap W \neq \emptyset \mid X = x)}.$$

return function $C^{\mathrm{gr}}_\eta : \mathcal{X} \times [0, 1] \rightrightarrows \mathcal{Y}$ defined by

$$C^{\mathrm{gr}}_\eta(x, u) \coloneqq \begin{cases} C^{\mathrm{gr},j(x,\eta)}(x) & \text{if } u < t_\eta(x), \\ C^{\mathrm{gr},j(x,\eta)-1}(x) & \text{otherwise.} \end{cases}$$

Alg. 2 returns a nested sequence $\{C^{\mathrm{gr}}_\eta(x, U)\}_{\eta \in (0,1)}$, where $U \sim \mathsf{Uni}[0, 1]$ randomizes to achieve an appropriate level. While the sequence need not necessarily solve problem (Cond) (see Example 1 to come), there are natural sufficient conditions for Algorithm 2 to return a size-optimal set, of which we present two. As the first particular case, consider that conditional on x, labels $y \in \mathcal{Y}$ belong to W independently:

Definition 13 *A probability distribution P on $W \in 2^{\mathcal{Y}}$ has* label-independent structure *if $\{1\{y \in W\}\}_{y \in \mathcal{Y}}$ are independent random variables when $W \sim P$.*

We might expect W to exhibit label independence when all labels $y \in \mathcal{Y}$ satisfy $\pi(y \mid x) \ll 1$, with the exception of a single label $y^\star(x)$, for which $\pi(y^\star(x) \mid x) \approx 1$, as will often be the case in low-noise classification settings.

Another scenario occurs when the label space exhibits a hierarchical tree structure, as one may expect in image classification (Deng et al., 2009) or structured prediction tasks (Cabannes et al., 2020). When the weak sets W obey the same structure the distribution—they are subtrees of the global tree—we say the labels have a tree structure (see Figure 1):

Definition 14 *A probability distribution P on $W \in 2^{\mathcal{Y}}$ has a* tree structure *if for all $w_1, w_2 \subset \mathcal{Y}$,*

$$P(W = w_1) > 0 \text{ and } P(W = w_2) > 0 \text{ imply } w_1 \cap w_2 \in \{w_1, w_2, \emptyset\}. \tag{8}$$

Both definitions (independent labels and hierarchically-structured weak labels) are sufficient to guarantee size-optimality for the greedy confidence sets Algorithm 2 constructs. The next Proposition, whose proof we provide in Appendix B.2.2, makes this formal.

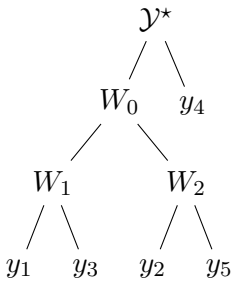

Figure 1. A tree-structured (8) distribution for W given $X = x$, with $\mathcal{Y}^\star = \{1, 2, 3, 4\}$. The possible configurations for W are the singletons $\{y_1\}, \{y_2\}, \{y_3\}, \{y_4\}$, the two pairs $W_1 = \{y_1, y_3\}$ and $W_2 = \{y_2, y_5\}$, $W_0 = \{y_1, y_2, y_3, y_5\}$, and $\mathcal{Y}^\star$ itself.

Proposition 15 *Suppose the probability law $\mathcal{L}(W \mid X = x)$ has either label-independent structure (Def. 13) or a tree structure (Def. 14). Then for all $\eta \in (0, 1)$, C_η^{gr} is conditionally size-optimal, and therefore is a minimizer in equation* (Cond).

In general, even with perfect knowledge of the distribution of $W \mid X = x$, the nested greedy confidence sets C_η^{gr} need not be size-optimal, as there may be weak sets appearing with high probability while their constituents do not, so that the conditionally size-optimal sets $\{C_\eta^{\mathrm{cond}}\}_{\eta \in [0,1]}$ are not nested. The next example illustrates one such failure mode:

Example 1 *Let the distribution of W be*

$$W = \begin{cases} \{1, 2\} & w.p.\ 0.3 \\ \{1, 3\} & w.p.\ 0.25 \\ \{2\} & w.p.\ 0.2 \\ \{3\} & w.p.\ 0.15 \\ \{1\} & w.p.\ 0.1. \end{cases}$$

Then for $\eta = 0.9$, it is immediate that $C_\eta^{\mathrm{cond}}(x, u) = \{2, 3\}$, but $C_\eta^{\mathrm{gr}}(x, u) = \{1, 2, 3\}$ or $\{1, 2\}$ depending on whether $u < 1/3$. In addition, $C_{eta'}^{\mathrm{cond}}(x, u) = \{1, 3\}$ when $\eta' = 0.85$, showing that in this case, the confidence set mappings C_η^{cond} need not be nested.

We show in Appendix A that even in general cases as example 1, the sizes of the confidence sets C^{gr} and C^{cond} cannot be too far apart.

4. Efficient conformalization for large output spaces

While Section 3 provides a generic treatment on for producing scoring functions and associated confidence sets of minimal size, in typical practice, a (pre-trained) model provides a predictive scoring function, which may not be directly associated to a probability metric, and we wish to leverage such models. This is of particular interest when the label space $\mathcal{Y}$ is large, as in structured prediction problems (Taskar, 2005; Cabannes et al., 2020), where computational efficiency becomes a main challenge. In this section, we thus first introduce a general method for computing and representing confidence set mappings of the form $\{y \mid s(x, y) \le \hat{t}_n\}$, and then describe how to efficiently carry out Alg. 1 in ranking (Section 4.2) and matching (Appendix C.1) problems.

4.1 Conformal confidence sets with sequential partitioning

We seek to efficiently compute and represent the confidence set $\widehat{C}_n(x)$ for any instance $x \in \mathcal{X}$, typically for a task where the label space contains more configurations than are efficiently enumerable ($K!$ for matching and ranking problems over K items). At the same time, recalling that $\widehat{t}_n$ denotes the threshold Algorithm 1, if our confidence sets are to be informative they should include relatively few configurations $y \in \mathcal{Y}$ satisfying $s(x, y) \leq \widehat{t}_n$. To the end of computing the set $\widehat{C}_n(x) = \{y \mid s(x, y) \leq \widehat{t}_n\}$ in Alg. 1, we focus on methods for computing a given number M of configurations with the smallest score $s(x, y)$. This is essentially without loss of generality: while we may not know the appropriate $M = M_x = |\widehat{C}_n(x)|$ to guarantee coverage, if for each $M \in \mathbb{N}$ we can find the M best configurations in time polynomial in M, then by sequentially doubling M until we obtain an element $y \in \mathcal{Y}$ such that $s(x, y) > \widehat{t}_n$, we achieve time polynomial in M_x. Algorithm 3 builds on this intuition to return a valid confidence set. The Algorithm we suggest is essentially an extension of the algorithm proposed by Chegireddy and Hamacher (1987) to find the K-best matchings in a bipartite graph. We reuse the general idea (i.e. compute a sequence of partitions of the space and maintain a list of the two best configurations for each item of the partition) and extend it to a more general structured prediction. The key is to observe that we only need to be able to compute the two best configurations of a given subset of configurations, which they do on matching problems (and our Algorithm essentially reduces to theirs in the matching case). We then apply that paradigm to the ranking case.

We remark briefly that an alternative approach is to conformalize directly on the size M of the confidence set: suppose we learn a function $\widehat{M} : \mathcal{X} \to \mathbb{N}$ predictive of the rank (according to $\{s(x, y)\}_{y \in \mathcal{Y}}$) of the first "compatible" configuration, i.e predictive of

$$M_i \coloneqq \text{rank of the first configuration } y \in \mathcal{Y} \text{ such that } \varphi_i(y) = Y_i^{\text{weak}}.$$

In that case, if we let $\widehat{Q}_n \coloneqq \left(1 + n^{-1}\right)(1 - \alpha)$-quantile of $\{M_i - \widehat{M}(X_i)\}_{i=1}^n$, we would only need to return

$$\widehat{C}_n(x) \coloneqq \left\{\widehat{M}(x) + \widehat{Q}_n \text{ best configurations } y \in \mathcal{Y} \text{ ordered by } s(x, y)\right\}.$$

This approach makes prediction more efficient (as we know in advance the number of configurations to compute), but the computational effort of the conformalization step (5) increases, as we must compute the rank of the best constrained configuration for each instance.

4.1.1 Returning M best configurations with sequential partitioning

Let us now fix $M \geq 1$, and focus on retrieving the M configurations with the lowest scores. Algorithm 3 provides a general recipe using dynamic programming, and it is efficient as long as we can efficiently compute certain partitions of the label space. We require the following definition.

Definition 16 *A function* Partition : $2^{\mathcal{Y}} \times \mathcal{Y} \times \mathcal{Y} \to 2^{\mathcal{Y}} \times 2^{\mathcal{Y}}$ *is* valid for a score function s *if, for every subset* $\tilde{\mathcal{Y}} \subset \mathcal{Y}$ *and pair of configurations* $y_1, y_2 \in \tilde{\mathcal{Y}}$ *satisfying*

$$y_1 \in \operatorname*{argmin}_{y \in \tilde{\mathcal{Y}}} s(x, y) \quad \textit{and} \quad y_2 \in \operatorname*{argmin}_{y \in \tilde{\mathcal{Y}} \setminus \{y_1\}} s(x, y),$$

Partition$(\tilde{\mathcal{Y}}, y_1, y_2)$ *returns a partition* $(\tilde{\mathcal{Y}}_1, \tilde{\mathcal{Y}}_2)$ *of* $\tilde{\mathcal{Y}}$ *such that* $y_1 \in \tilde{\mathcal{Y}}_1$ *and* $y_2 \in \tilde{\mathcal{Y}}_2$.

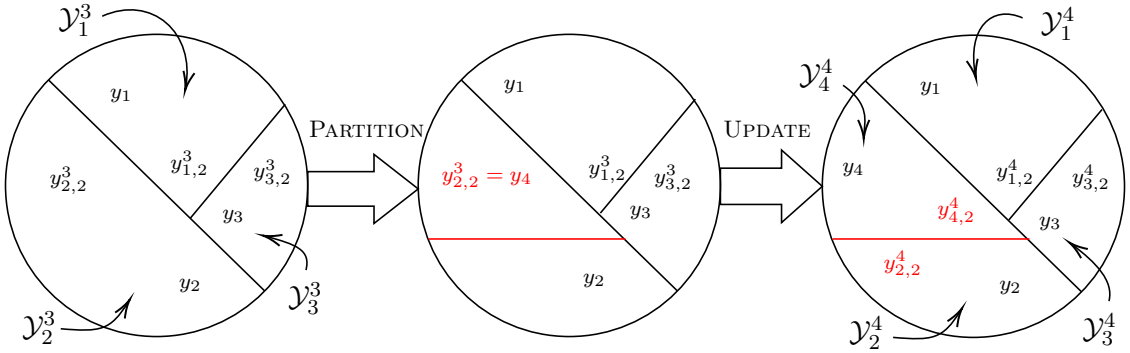

Figure 2. Alg. 3 scheme for sequential partitioning: first, partition the subset containing the $m+1$-th best configuration, $y_{2,2}^3$ in this case, then compute both second-best configurations in the newly formed subsets of the partition—here $\mathcal{Y}_2^4$ and $\mathcal{Y}_4^4$.

We thus leverage two conditions: a valid Partition for our score function s and, for each pair of subsets $\mathcal{Y}_1, \mathcal{Y}_2 \subset \mathcal{Y}$ that it produces, we must be able to (efficiently) compute the second-best configurations in $\mathcal{Y}_1$ and $\mathcal{Y}_2$, i.e.,

$$y_{1,2} \in \operatorname*{argmin}_{y \in \mathcal{Y}_1 \setminus \{y_1\}} s(x,y) \quad \text{and} \quad y_{2,2} \in \operatorname*{argmin}_{y \in \mathcal{Y}_2 \setminus \{y_2\}} s(x,y).$$

Figure 2 encapsulates the main idea Alg. 3: at each step $m \in [M]$, we maintain a partition $\{\mathcal{Y}_j^m\}_{j=1}^m$ of $\mathcal{Y}$ such that if $y_j^m \in \operatorname{argmin}_{y \in \mathcal{Y}_j^m} s(x,y)$, then for all $j \in [m]$, we have

$$y_j^m \in \operatorname*{argmin}_{y \in \mathcal{Y} \setminus \{y_1^m, \ldots, y_{j-1}^m\}} s(x,y),$$

i.e., y_j^m is the j-th best configuration in $\mathcal{Y}$. Now, for each $j \in [m]$, let the configuration $y_{j,2}^m \in \operatorname{argmin}_{y \in \mathcal{Y}_j^m \setminus \{y_j^m\}} s(x,y)$ be the second-best configuration in $\mathcal{Y}_j^m$. The key is then to observe that if we set

$$\operatorname{ind}(m) := \operatorname*{argmin}_{j \in [m]} s(x, y_{j,2}^m),$$

then $y_{\operatorname{ind}(m),2}^m$ is the $(m+1)$st best configuration in $\mathcal{Y}$. The Partition function then divides $\mathcal{Y}_{\operatorname{ind}(m)}^m$ into two sets $\mathcal{Y}_{\operatorname{ind}(m)}^{m+1}$ and $\mathcal{Y}_{m+1}^{m+1}$ such that $y_{\operatorname{ind}(m)}^m \in \mathcal{Y}_{\operatorname{ind}(m)}^{m+1}$ and $y_{\operatorname{ind}(m),2}^m \in \mathcal{Y}_{m+1}^{m+1}$. Under the assumption that Partition is valid (Def. 16) for the score s, the following lemma guarantees the validity of Algorithm 3.

Lemma 17 *Assume the* Partition *function is valid for the score function s. Then Algorithm 3 returns a set of configurations $\{y_j\}_{j=1}^M$ such that for each $j \in [M]$,*

$$y_j \in \operatorname*{argmin}_{y \in \mathcal{Y} \setminus \{y_1, \ldots, y_{j-1}\}} s(x,y).$$

Proof This follows by an induction over $m \geq 1$, which guarantees that at every step $m \geq 1$, $\{\mathcal{Y}_j^m\}$ is a partition of $\mathcal{Y}$ such that $y_j^m = \operatorname{argmin}_{y \in \mathcal{Y}_j^m}$ and

$$s(x, y_1^m) \leq s(x, y_2^m) \cdots \leq s(x, y_m^m) \leq \min_{y \in \mathcal{Y} \setminus \{y_j^m\}} s(x,y).$$

Algorithm 3 Sequential partitioning

Require: score function $s : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$; valid (Def. 16) Partition: $2^{\mathcal{Y}} \times \mathcal{Y} \times \mathcal{Y} \to 2^{\mathcal{Y}} \times 2^{\mathcal{Y}}$; instance $x \in \mathcal{X}$

initialize: Compute $y_1^1 \coloneqq \operatorname{argmin}_{y \in \mathcal{Y}} s(x, y)$ and $y_{1,2}^1 \coloneqq \operatorname{argmin}_{y \in \mathcal{Y} \setminus \{y_1^1\}} s(x, y)$.

Set $\mathcal{Y}_1^m \coloneqq \mathcal{Y}$ {Initialize the partition}

for $m = 1, 2, \ldots, M-1$ **do**

  $\mathrm{ind}(m) \coloneqq \operatorname{argmin}_{j \in [m]} s(x, y_{j,2}^m)$ {Find the $m+1$-th best configuration}

  $y_{m+1}^{m+1} \coloneqq y_{\mathrm{ind}(m),2}^m$

  for $j \in [m] \setminus \{\mathrm{ind}(m)\}$ **do**

    $(\mathcal{Y}_j^{m+1}, y_j^{m+1}, y_{j,2}^{m+1}) \coloneqq (\mathcal{Y}_j^m, y_j^m, y_{j,2}^m)$ {All subsets $\{\mathcal{Y}_j^m\}_{j \neq \mathrm{ind}(m)}$ remain identical}

  end for

  $\mathcal{Y}_{\mathrm{ind}(m)}^{m+1}, \mathcal{Y}_{m+1}^{m+1} \coloneqq \textsc{Partition}(\mathcal{Y}_{\mathrm{ind}(m)}^m, y_{\mathrm{ind}(m)}^m, y_{m+1}^{m+1})$ {Partition the set $\mathcal{Y}_{\mathrm{ind}(m)}^m$}

  $y_{\mathrm{ind}(m)}^{m+1} \coloneqq y_{\mathrm{ind}(m)}^m$ and $y_{\mathrm{ind}(m),2}^{m+1} \coloneqq \operatorname{argmin}_{y \in \mathcal{Y}_{\mathrm{ind}(m)}^{m+1} \setminus \{y_{\mathrm{ind}(m)}^{m+1}\}} s(x, y)$

  $y_{m+1,2}^{m+1} \coloneqq \operatorname{argmin}_{y \in \mathcal{Y}_{m+1}^{m+1} \setminus \{y_{m+1}^{m+1}\}} s(x, y)$ {Compute second-best configurations}

end for

return $\{y_m^M\}_{m=1}^M$

The property transitions from m to $m+1$ as the Partition function is valid, and we choose y_{m+1}^{m+1} as the best second-best configuration, hence it is the $(m+1)$st best configuration. ■

The existence of an efficient valid Partition function is instance-dependent and typically requires a specific choice of scoring function; we provide concrete implementations for two types of structured prediction problems.

4.2 Structured prediction examples (Ranking problems and partial labeling mechanisms)

While Algorithm 3 is generic, we now show that efficient partitioning and minimization functions exist in structured prediction instances, so that we may efficiently carry out above algorithms in the instance. Here we focus on ranking problems and defer the discussion on matching tasks in Appendix C.1.

The goal here is to predict a preference ranking $y \in \mathcal{Y} = \mathfrak{S}_K$ of K different items, documents, for a certain user or query $x \in \mathcal{X}$, where $y(i)$ denotes the item of rank i. Typically, one achieves this by learning relevance functions $r_k : \mathcal{X} \to \mathbb{R}$, which evaluate each item $1 \leq k \leq K$ individually before aggregating into a single ranking prediction (Freund et al., 2003; Duchi et al., 2013; Qin and Liu, 2013; Cao et al., 2007). We assume here that we have access to such relevance functions.

In ranking tasks, there are two reasonable ways in which practitioners may acquire partial supervision or user feedback. The first mechanism (Cabannes et al., 2020) assumes they only receive a subset of all $\binom{K}{2}$ pairwise comparisons $\left(1\{y^{-1}(i) < y^{-1}(j)\}\right)_{1 \leq i < j \leq K}$ as a partial label, which is especially relevant in cases where the practitioner solicits feedback from users by asking them to compare a small number of items. Unfortunately, carrying out the computation (5) in Alg. 1 is an NP-hard problem, namely the minimal cost feedback

arc set problem (Ailon et al., 2008; van Zuylen et al., 2007), for which only an approximate solution is available (by solving an integer linear program).

Another form of feedback, on which we focus in the rest of the section and that allows to run both Algs. 1 and 3 efficiently, instead assumes that users only provide a fraction of their preferred ranking and reveal $(y(i))_{i=1}^{K^{\text{partial}}}$ for some $K^{\text{partial}} \leq K$. In order to construct score functions suitable to the application of Alg. 3, we first introduce ranking-consistent score functions.

Definition 18 *A scoring function s^{rank} is* ranking-consistent *with a set of relevance functions $\{r_k : \mathcal{X} \to \mathbb{R}\}_{k\in[K]}$ if a for all $1 \leq i < j \leq K$ and $y \in \mathfrak{S}_K$,*

$$s^{\text{rank}}(x, (i, j) \circ y)) \leq s^{\text{rank}}(x, y) \text{ if } r_{y(i)}(x) \leq r_{y(j)}(x), \tag{9}$$

where $(i, j) \circ y$ denotes transposition of i and j in the permutation y.

Such a scoring mechanism should always favor a ranking that gives a higher rank to $y(j)$ than $y(i)$ if $r_{y(i)}(x) \leq r_{y(j)}(x)$, i.e., if $y(j)$ has a greater relevance than $y(i)$. It ensures in particular that the $(m + 1)$st best ranking is always a "neighbor" of one of m best; this is an immediate property of the score function, as it can always increase by swapping two elements i and j that are mis-ordered.

An example of ranking-consistent scoring function is the disagreement-based scoring function (Kendall, 1938; Kemeny, 1959; Duchi et al., 2013)

$$s^{\text{rank}}(x, y) \coloneqq \sum_{i<j} \psi\left(r_{y(i)}(x), r_{y(j)}(x)\right), \tag{10}$$

where $\psi : \mathbb{R}^2 \to \mathbb{R} \geq 0$ is a function satisfying $\psi(a, b) = 0$ when $a \geq b$ and $\psi(a, b) > 0$ when $a < b$, non-increasing in the first argument and non-decreasing in the second. Unless we specify otherwise we use $\psi(a, b) = [b - a]_+$ in our experiments.

Finding the configuration y that minimizes the partial score (5) of a ranking-consistent score function is straightforward: it suffices to rank all the elements in $[K] \setminus \{y(i)\}_{i=1}^{K^{\text{partial}}}$ according to their relevance scores $(r_j(x))_{j=1}^{K}$, and then append them to the first K^{partial} elements. More interestingly, this property allows efficiently retrieving the $M \geq 1$ best configurations with Alg. 3. Throughout the loop, we make sure that any set of permutations $\mathcal{Y}_j^m$ is a subset of permutations consistent with a finite number of partial rankings (pairwise comparisons), and that its best and second-best configurations y_j^m and $y_{j,2}^m$ only differ by a neighboring transposition of the form $(i + 1, i)$, satisfying

$$y_{j,2}^m \coloneqq \operatorname*{argmin}_{y\in\mathcal{Y}_j^m} \{s(x, y) \mid \exists i \in [K],\ y = (i + 1, i) \circ y_j^m\}. \tag{11}$$

If we can guarantee this loop invariant, then there always exists $i_{j,m} \in [K]$ such that $y_{j,2}^m = (i_{j,m} + 1, i_{j,m}) \circ y_j^m$, and we only need to define the partition function on a smaller subset of $2^{\mathcal{Y}} \times \mathcal{Y} \times \mathcal{Y}$: for any subset of permutations $\tilde{\mathcal{Y}} \subset \mathcal{Y}$, $\tilde{y} \in \tilde{\mathcal{Y}}$ and $i \in [K]$ such that $(i + 1, i) \circ \tilde{y} \in \tilde{\mathcal{Y}}$, we let

$$\begin{aligned}&\textsc{Partition}_{\text{Ranking}}(\tilde{\mathcal{Y}}, \tilde{y}, (i + 1, i) \circ \tilde{y}) \coloneqq \\ &\quad \tilde{\mathcal{Y}} \cap \left\{y \in \mathcal{Y} \mid y^{-1}(\tilde{y}(i)) < y^{-1}(\tilde{y}(i + 1))\right\}, \tilde{\mathcal{Y}} \cap \left\{y \in \mathcal{Y} \mid y^{-1}(\tilde{y}(i)) < y^{-1}(\tilde{y}(i + 1))\right\},\end{aligned} \tag{12}$$

splitting $\tilde{\mathcal{Y}}$ according to whether $\tilde{y}(i)$ has a higher rank than $\tilde{y}(i+1)$.

The next lemma, whose proof is in Appendix B.3.1, states that this partition rule indeed guarantees that, at every step m of the loop in Algorithm 3, the second-best configuration in $\mathcal{Y}_j^m$ satisfies the invariant (11).

Lemma 19 *Assume the score function is ranking-consistent* (9) *for a set of relevance functions* $\{r_k\}_{k=1}^K$. *Then Algorithm 3 with the* $\textsc{Partition}_{Ranking}$ *function* (12) *produces a sequence of partitions with second-best configurations satisfying equation* (11).

That is, Algorithm 3 is correct.

5. Experiments

In this section, we test our weakly supervised methods experimentally, in different classification and regression problems, on both synthetic and real datasets, with an emphasis on their computational efficiency and informativeness. Here we focus on ranking problems and present further experiments on matching and regression problems in the Appendix C. The primary goal of this paper is not to provide end-to-end models with only partially supervised data, but rather to introduce a new form of coverage validity and show how to achieve it with partially labeled data. In contrast to the split-conformal method (Vovk et al., 2005), which requires fully supervised instances for both training and validating, we only need these to train a model and form a scoring function suitable for the application of Alg. 1. In some cases, standard models already exist, such as in image classification (He et al., 2016).

To provide a meaningful comparison with existing conformal methods and test for predictive set size efficiency, we use fully labeled real datasets, and introduce different plausible forms of weak supervision on our calibration and test sets before applying Algorithm 1 to construct confidence sets. Our method displays similar behavior across all datasets and forms of partial information. To provide a baseline, we also run a standard fully supervised conformal scheme (FSC) using the strong labels Y_i and true scores $s(X_i, Y_i)$, which runs similarly as Alg. 1, but with threshold

$$\widehat{t}_n^{\text{full}} \coloneqq (1+n^{-1})(1-\alpha)\text{-quantile of } \{s(X_i, Y_i)\}_{i=1}^n. \tag{13}$$

We can then estimate the gain in efficiency—in the form of decreased confidence set sizes—that stems from the weakening of strong coverage (1) to weak coverage (3).

5.1 A toy classification example

We first perform an experiment with a toy multiclass data set containing $K = 10$ different classes and $d = 2$ dimensional features. We consider a partially supervised problem on $\mathcal{X} \times \mathcal{Y} = \mathbb{R}^d \times [K]$ for which we wish to output valid confidence sets. We use the following model: each potential response $y \in [K]$ has a noisy score depending on the feature vector $X \in \mathbb{R}^d$ though a vector $\theta_y^\star \in \mathbb{R}^d$,

$$\{S_y^{\text{oracle}}\}_{y\in[K]} \mid X = x \sim \mathsf{N}(\{x^T\theta_y^\star\}_{y\in[K]}, \sigma^2 I_K) \tag{14}$$

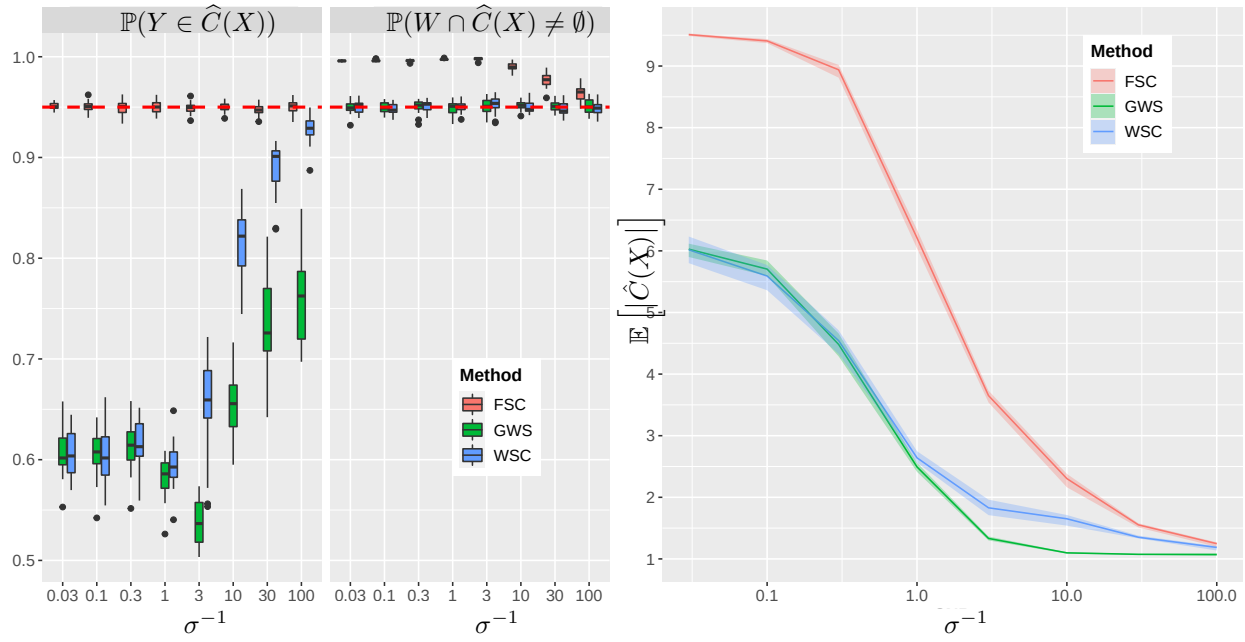

Figure 3. Results for the simulated multiclass data (14), over $N_{\text{trials}} = 20$ trials. The left plot shows respectively the strong (1) and weak (3) coverage for the greedy weakly supervised (GWS), the weakly supervised conformal (WSC) and the fully supervised conformal (FSC) confidence sets. The right plot displays the average confidence set size for these methods.

Ideally, we would recover the strong label $Y \coloneqq \operatorname{argmin}_{y \in \mathcal{Y}} S_y^{\text{oracle}}$, but our weakly supervised methods do not observe Y directly: instead, for a random instance-dependent threshold T, we only have access to the weak set

$$W \coloneqq \{y \in \mathcal{Y} \mid S_y^{\text{oracle}} \leq T\}.$$

As motivation, consider a supervised learning task in which, out of all potential responses, there is always only one ground truth, but there are other labels that are "good enough" (i.e. have a low enough score) to answer a certain query. In this setting, a confidence set is weakly valid (3) as long as it contains at least one label y such that $S_y^{\text{oracle}} \leq T$, whereas it is strongly valid (1) if it contains Y.

We vary the signal-to-noise ratio $\sigma^{-1} \in \{10^{-2}, \ldots, 10^2\}$: when it is too small, no model (even an oracle one) can be highly predictive, and a standard conformal method should provide large uninformative confidence sets, whereas we expect our new definition of coverage to yield smaller sets, as any label in W (i.e. with a low enough score) provides valid coverage.

In this experiment, we compare three different methods. The "Greedy weakly supervised" (GWS) method only uses partially labeled data both when training and conformalizating. It first trains K separate logistic regressions with $\{X_i\}$ as features and each $\{1\{y \in W_i\}\}$ for all $y \in \mathcal{Y}$ as potential response, providing a model for $P(y \in W \mid X = x)$, and models the distribution of W given $X = x$ as label-independent (see Definition 13). It then computes a nested sequence of confidence sets thanks to Alg. 2, which we then feed to the conformalization Algorithm 1 using the nested scoring mechanism (6).

The second and third methods, the "Weakly supervised conformal" (WSC) and the "Full supervised conformal" (FSC) methods respectively, use fully supervised data for training: we first train a standard logistic regression model $p_\theta(y \mid x) \propto \exp(\theta_y^T x)$ on $\{(X_i, Y_i)\}$, and then construct a scoring function using the Generalized Inverse Quantile (GIQ) procedure that Romano et al. (2020) introduce. In the conformalization step, the WSC method runs Algorithm 2 with partially labeled calibration data, while the FSC method uses strongly labeled data to compute the threshold $\widehat{t}_n^{\text{full}}$ in (13). The threshold $\widehat{t}_n$ in Alg. 2 is always smaller than $\widehat{t}_n^{\text{full}}$, so the FSC method returns larger confidence sets than the WSC method. We expect that as the signal to noise ratio decreases, the gap between the GWS and WSC confidence sets and the FSC confidence sets increases.

The precise experimental set-up is as follows: we simulate $n = 10^4$ data points, splitting them into training (30%), calibration (20%) and test (50%) sets. We draw each θ_y uniform on $\mathbb{S}^{d-1}$, $\{X_i\}_{i=1}^n \overset{\text{iid}}{\sim} \mathsf{N}(0, I_d)$, choosing weak threshold $T \sim \mathsf{Uni}[\min_{y\in\mathcal{Y}}\{S_y^{\text{Oracle}}\}, \max_{y\in\mathcal{Y}}\{S_y^{\text{Oracle}}\}]$. We repeat the entire process $N_{\text{trials}} = 20$ times to account for uncertainty, presenting our results in Figure 3.

As we expect, using an alternative weaker version of coverage (3) allows us to significantly decrease the size of the confidence set (by up to a factor of 3), especially when the signal-to-noise ratio is small, as one must include more classes in the confidence set to maintain strong coverage. Indeed, we can see that the strong coverage (1) for the GWS and WSC procedures fall well below $1-\alpha = 95\%$ in this case, since they only strive for weak $1-\alpha$ coverage, which they consistently achieve. Since the GWS method aims to construct minimal confidence sets, we expect that it produces smaller confidence sets than the WSC method, which simply leverages an existing strongly supervised model; we consistently observe this across different values of σ.

5.2 Document ordering for query answering

We now present the results of two experiments using Alg. 1 in a ranking problem. The first one is a simulation of a standard ranking task, while the second focuses on ranking documents' relevance to specific queries in the Microsoft LETOR dataset (Qin and Liu, 2013).

5.2.1 Ranking simulation study

In a first simulation study, we aim to predict a ranking of labels $y \in [K]$ based on a feature vector $X \in \mathbb{R}^d$. Think here of a supervised problem where we want to rank users' preferences for a set of items. Each user has an unknown relevance score $S_y^{\text{Oracle}} \in \mathbb{R}$ for each item $y \in [K]$, which induces a ground truth ranking over the labels:

$$Y := \text{argsort}\{S_y^{\text{Oracle}}\}_{y\in[K]} \in \mathfrak{S}_K.$$

The problem is to recover this noisy ranking and produce valid confidence sets in $\mathfrak{S}_K$, but our weakly supervised methods do not observe the full ranking when conformalizing: they can only observe the ranking up to the $K_{\text{partial}} \leq K$-th element, leading to the weak set

$$W = \{y \in \mathfrak{S}_K \mid \ \forall j \in [K_{\text{partial}}], y(j) = Y(j)\}. \tag{15}$$

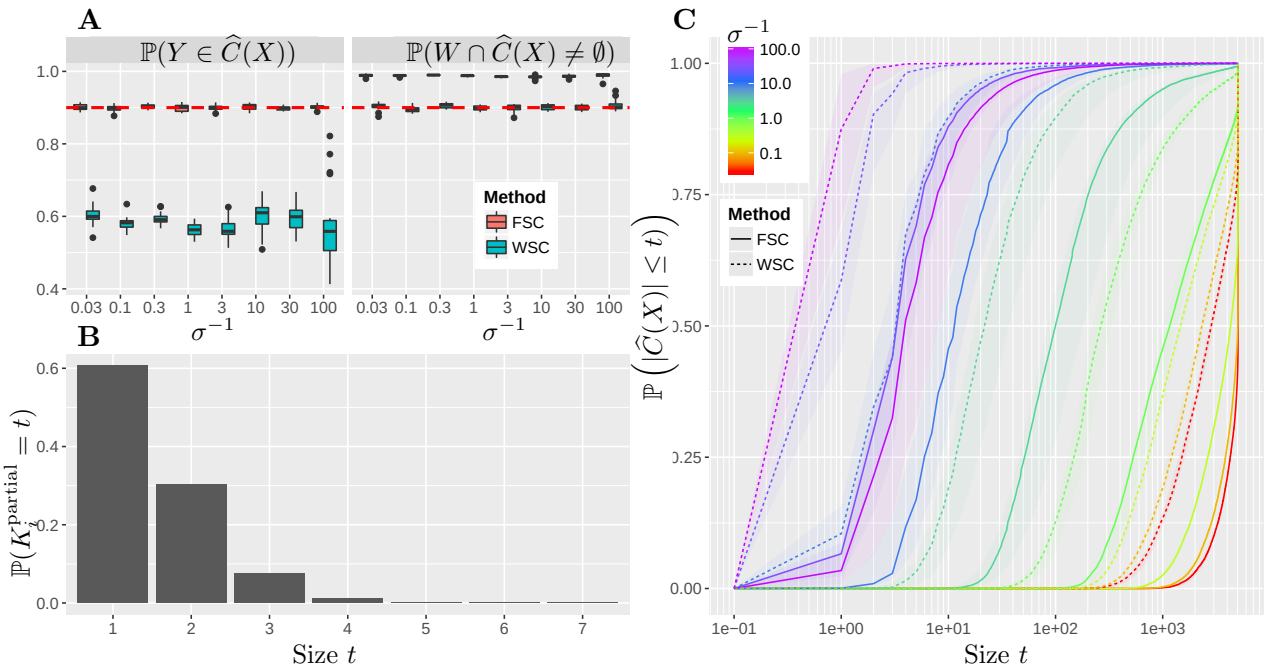

Figure 4. Results for the ranking simulation study (15) over $N_{\text{trials}} = 20$ trials. A: Strong (1) and Weak (3) coverage for the weakly supervised conformal (WSC) and the fully supervised conformal (FSC) confidence sets. B: Density histogram of the variable K^{partial} (15) governing the weak distribution in this example. C: Distribution of the confidence set size $|\widehat{C}(X)|$ for different signal-to-noise ratios σ^{-1}.

In our experiment, we simulate $n = 10^4$ i.i.d. different users, using the same (30,20,50) train/validation/test split as in Section 5.1. With $K = 7$ and $d = 2$, we draw the user feature vector $X_i \overset{\text{iid}}{\sim} \mathsf{N}(0, I_d)$, and then conditionally on X_i, we produce normal item-wise relevance scores $\{S_{iy}^{\text{Oracle}}\}_{i\in[n],y\in[K]}$ following the distribution (14). We finally simulate partial supervision by drawing the number of observed elements in the ranking K_i^{partial} as $\min(K, 1 + A_i)$, where $A_i \overset{\text{iid}}{\sim} \text{Poi}(.5)$. The lower left panel of Figure 4 shows the overall distribution of this quantity: most users only reveal the first 1 to 3 items in their optimal ranking.

We then produce strongly and weakly valid confidence sets at the $1 - \alpha \coloneqq 90\%$ level. We use the same scoring model for both the fully supervised conformal (FSC) and weakly supervised conformal (WSC) procedures: we learn linear individual relevance score functions $\{r_y\}_{y\in[K]}$ (with fully supervised training data) via the ListNet procedure (Cao et al., 2007), which we briefly describe here. Given a set of relevance scores $\{r_y\}_{y\in\mathcal{Y}} \in \mathbb{R}^K$, ListNet models the probability of a ranking $\pi \in \mathfrak{S}_K$ as

$$P_r(\pi) \coloneqq \prod_{y=1}^{K} \frac{\exp(r_{\pi(y)})}{\sum_{l=y}^{k} \exp(r_{\pi(l)})}, \tag{16}$$

which gives each item $y \in \mathcal{Y}$ a top-1 probability (of ranking first) equal to

$$P_r^1(y) \coloneqq P_r(\pi(1) = y) = \frac{\exp(r_y)}{\sum_{l=1}^{k} \exp(r_l)}.$$

Given a training data set containing pairs $(X, R) \in \mathcal{X} \times \mathbb{R}^K$ of features/relevance scores, we learn score mappings by minimizing the log-loss of the top-1 distribution over a set $\mathcal{F}$ of functions

$$\{\hat{r}_y\}_{y\in[K]} \coloneqq \underset{\tilde{r}\in\mathcal{F}^{\mathcal{Y}}}{\text{argmin}} \left\{ \sum_{(X,R)\in \text{ training data}} \sum_{k=1}^{K} -P_R^1(k) \log\left(P_{\tilde{r}(X)}^1(k)\right) \right\}.$$

In our experiment, we only observe the ranking (or even a fraction of), not the true per-item relevance scores, hence, following common practice (Cao et al., 2007), we use $R_y = K -$ the rank of the item $= K - Y^{-1}(y)$ as a proxy for our observed item-wise relevance scores when training our model.

In our experimental set up, each relevance score function $r_y : \mathcal{X} \to \mathbb{R}$ ideally estimates the true conditional mean of the oracle scores, $x \mapsto x^T\theta_y^\star$. Given these individual scores, we use the scoring mechanism (10) with $\psi(x, y) \coloneqq (y - x)_+$ and conformalize using the strategy we describe in Section 4.2.

The difference between the WSC and FSC methods is the conformalization step on calibration data: WSC runs Alg. 1 with partially supervised data to compute the score threshold $\widehat{t}_n$ while FSC uses strongly supervised data to return the more conservative threshold $\widehat{t}_n^{\text{full}}$ in (13).

Our results fit our initial expectations, in line with our first experiments: the size of the confidence set, as Figure 4C shows, benefits from the weaker definition of coverage: for any value of the signal to noise ratio $\sigma^{-1} > 0$, the WSC method produces much smaller

and more informative confidence sets than the FSC method, as it only needs to include a ranking with the correct first K_{partial} elements to provide valid coverage. ,At the cost of the strong coverage falling below $1-\alpha$ (see Fig. 4A), and with little information (only the first $K_{\text{partial}} < K$ labels), the WSC method constructs predictive sets that are much smaller and yet still valid (in a weak sense).

5.2.2 Ranking experiment with Microsoft LETOR dataset

We now tackle a slightly different type of ranking problem: we wish to rank a set of potential documents by order of relevance to a specific user query: documents more relevant to the query should occupy a higher position in the final ranking. A search engine is a good example of such problem: a user makes a search query, and the task is to sort Web pages that best answer that query among a (potentially large) set of potential pages.

Learning to rank with Microsoft LETOR dataset (Qin and Liu, 2013) To study that problem, we design an experiment with Microsoft LETOR data set. For each potential query/document pair (x, d), the dataset aggregates several quantities of interest to determine whether d is relevant to x into a $d = 46$-dimensional feature vector $\phi(x, d) \in \mathbb{R}^d$. For a query x with a set of potentially relevant documents $D(x) \coloneqq \{d_j\}_{j=1}^{|D(x)|}$, our data set additionally contains a ranking $\Pi(x) \in \mathfrak{S}_{|D(x)|}$ that orders these documents according to their relevance. Our goal is to retrieve that ranking using the feature vectors $\{\phi(x, d_j)\}_{j=1}^{D(x)}$.

A semi-synthetic weakly supervision set-up Here is how we construct weakly supervised calibration and test data sets. For each split (calibration/test), we first sample $n = 2000$ queries from the entire set of queries in LETOR validation and test datasets. For every query X_i, we select $K \in \{2, 4, 6, 8, 10, 20\}$ documents by first sorting $D(X_i)$ into K equally sized subsets by relevance, so subset $\ell \in [K]$ contains the documents with rank $\Pi(x)_y$ for every $y \in \{\frac{(\ell-1)|D(X_i)|}{K} + 1, \ldots, \frac{\ell|D(X_i)|}{K}\}$, and then drawing one document from each box uniformly at random.

This procedure ensures that there exists a significant relevance gap between any two potential documents in the query, and that the number of documents to rank is sufficient to allow reasonably-sized confidence sets. $\Pi(X_i)$ additionally induces a sub-ranking $Y_i \in \mathfrak{S}_K$ on these documents, which we treat as a strong label. Similarly to our approach in Section 5.2.1, we introduce partial labels by assuming that our weakly supervised method can only access the first K_i^{partial} elements of Y_i, where $K_i^{\text{partial}} \overset{\text{iid}}{\sim} 1 + \text{Poi}(.5)$: this simulates the plausible setting where a user has given feedback on the most relevant documents to the query, but certainly not to all of them. We repeat the entire simulation procedure $N_{\text{trials}} = 20$ times.

Building a ranking scoring function (10) We next describe how we use fully supervised training data to construct the scoring function that we feed Alg. 1 with. We first learn a linear query/document relevance function

$$r_\theta(x, d) \coloneqq \theta^T \phi(x, d) \tag{17}$$

using the ListNet procedure (16) on LETOR (fully supervised) train data

$$\left(x_i, (d_{i,j})_{j=1}^{D(x_i)}, y_i \in \mathfrak{S}_{D(x_i)}\right)_{i=1}^{n_{\text{train}}},$$

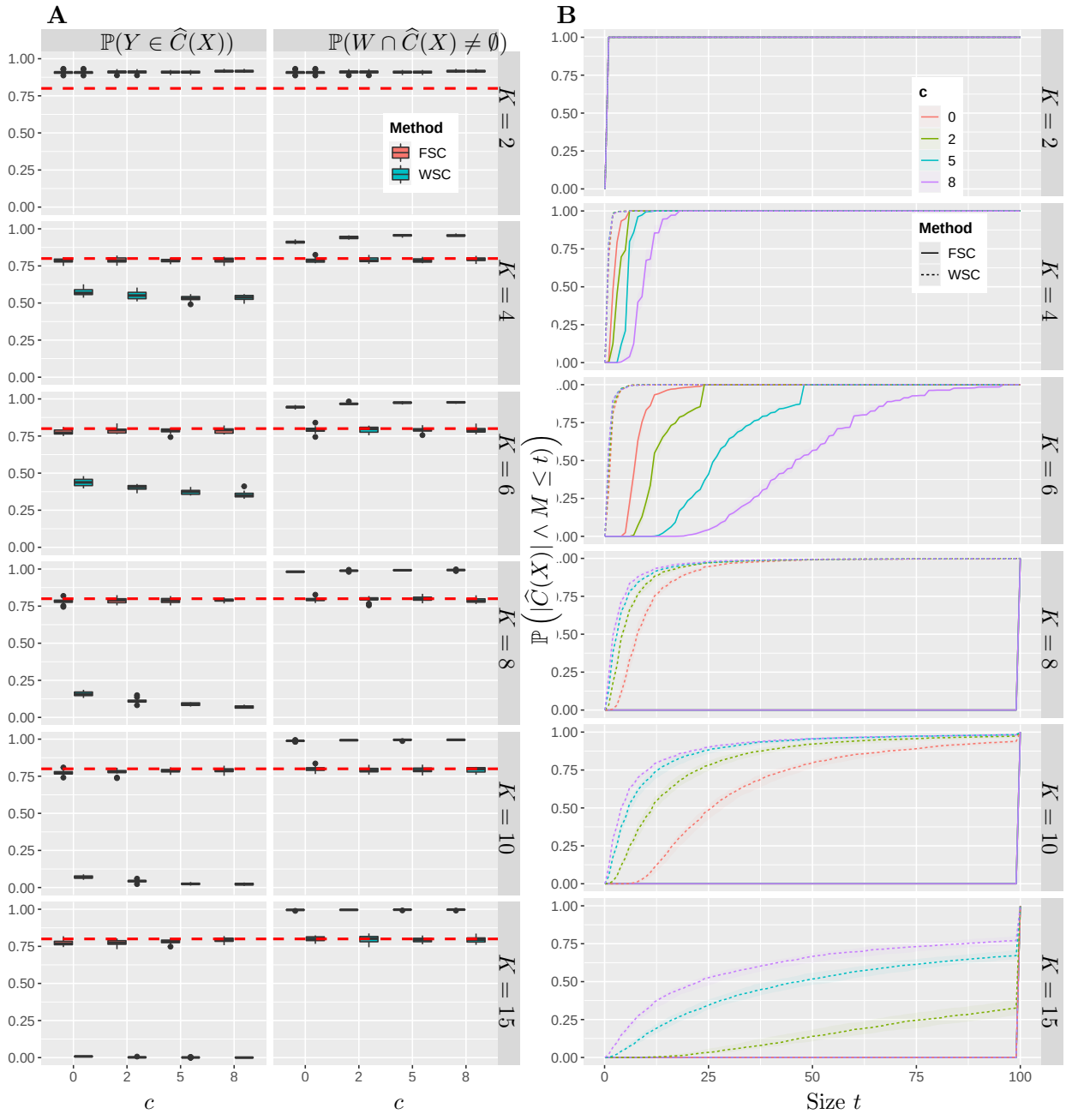

Figure 5. Results for LETOR ranking dataset (Qin and Liu, 2013) over $N_{\text{trials}} = 20$ trials. Each plot represents a different value of $K \in \{2, 4, 8, 10, 15\}$, the number of documents to rank, and we compare different scoring functions by varying the value of $c \in \{0, 2, 5, 8\}$ in equation (18). A: Strong (1) and Weak (3) coverage for the weakly supervised conformal (WSC) and the fully supervised conformal (FSC) confidence sets. B: Distribution of the confidence set size $|\widehat{C}(X)|$ for different numbers K of suggested documents. We display here the distribution of $\min(|\widehat{C}(X)|, M)$ for $M = 100$.

containing 55700 different query/document pairs.

We then use a specific implementation of the score function s^{Ranking} as in Eqn (10): if we rank K documents $\{d_k\}_{k\in[K]}$ for a query x, we rescale our relevance scores to the interval $[0,1]$,

$$\{r_k(x)\}_{k\in[K]} \coloneqq \left\{ \frac{r_\theta(x,d_k) - \min_{j\in[K]} r_\theta(x,d_j)}{\max_{j\in[K]} r_\theta(x,d_j) - \min_{j\in[K]} r_\theta(x,d_j)} \right\}_{k\in[K]},$$

and then, for a choice of $c \in \{0,2,5,8\}$, apply the scoring mechanism (10) with these relevance scores and pairwise comparison function

$$\psi_c(r_1,r_2) \coloneqq \exp(-cr_1)\,(r_2-r_1)_+\,. \tag{18}$$

In this example, we guarantee weak coverage if the true ranking Y_i on the first K_i^{partial} elements coincides with either one of the predictive rankings. To keep the predictive set size small, we thus wish to ensure that it doesn't contain two rankings with the same first K_i^{partial} elements (as they would be redundant): this is why we introduce the exponential term $\exp(-cr_1)$, which makes sure that when ranking all configurations by their score, highly ranked configurations have different first elements (rather than different last elements). To estimate the distribution of $|\widehat{C}(X)|$, we then compute the $M = 100$ best rankings for each query using Alg 3, and then replace the size of the confidence set by $\min(M, |\widehat{C}(X)|))$, effectively truncating it to M.

Experimental results We present our results in Figure 5. The confidence sets display the behavior we expect: when the number K of items to classify is small, the fully supervised conformal (FSC) and weakly supervised conformal (WSC) methods are similar, since partial labels are often equal to strong labels. Since the overall number of configurations is small, both methods are also able to maintain fairly small confidence sets. On the other hand, when K grows, the weak supervision method quickly departs from the full supervision one, and is able to produce confidence sets that are much smaller: when $K \geq 8$, the FSC method is unable to produce confidence sets with fewer than 100 configurations, as the number of configurations is large, and the problem is inherently noisy, especially for comparing documents with a fairly small relevance. The WSC (partially) overcomes that difficulty with its restrained notion of coverage, and is able to maintain a majority of confidence sets with size smaller than M, at least until $K = 15$. Of course, this method pays a price in terms of strong coverage, as for large K, the confidence set almost never contains the actual ground true ranking. That said, it may not a real issue as we are more interested in detecting which documents are actually relevant, and hence should have a higher rank, rather than correctly ordering documents with very little relevance to the query at the bottom of the list.

In addition, as we predicted, higher values of c in the pairwise comparison function (18) produce much smaller confidence sets by favoring more diverse rankings at the top of the list.

6. Discussion

The new measures of coverage we develop here—tailored to partially supervised data that may be easier to collect in many engineering and measurement-centric scientific scenarios—

help to bridge a gap between typical conformal predictive inference methods, which require expensive supervised data, and problems with partial supervision, whose typical focus is on prediction but not uncertainty quantification. Our hope is for this paper to open several avenues for future work. First, Algorithm 1 does not currently quantify the amount of coverage it provides conditionally on the query function, which essentially means in an item ranking framework that we do not know ahead of time whether we guarantee the top 2 or top 10 elements of the ranking to be correct. This occurs first because the query function is unknown ahead of time, and second because coverage (3) is marginal over the full randomness of the sample. Similarly to conformal inference extension works bridging the gap between marginal and conditional coverage, or between marginal and label-wise coverage, one potential goal is to adapt these methods and even out coverage conditionally on (plausible) query functions.

Our approach, grounded in conformal inference methods, acts as a wrapper around any black box machine learning models, providing valid coverage guarantees independent of model quality. However, it's crucial to note that poorly trained models may impact the efficiency of prediction sets, particularly when these sets are large due to factors like training data scarcity. Thus, efforts to mitigate overfitting and train high-quality models are paramount to ensuring the efficiency of our method's prediction sets. Scalability, while generally manageable with our methods for large datasets, presents challenges primarily during the conformalization step. Recognizing the absence of a one-size-fits-all solution, we've tailored scalable methods for each machine learning problem presented in the paper, aiming to effectively address diverse applications. Further, addressing these concerns in future works holds significant promise for advancing the field.

Additionally, the new definition (3) is intrinsically a 0-1 loss-based approach, in the sense that the confidence set $\widehat{C}_n$ either covers the weakly supervised set or fails. A natural initial extension is thus similarly to what Bates et al. (2021) propose in the strongly supervised case, recognizing that many structured prediction problems (e.g., segmentation tasks or multilabel problems) benefit from more subtle and granular loss functions. In the same vein, we present a few efficient choices of scoring mechanisms for structured prediction, which highlight the practicality and potential application of our general methodology; it seems quite plausible that more sophisticated scoring models could yield substantial improvements.

In our view, one of the more exciting potential applications of this work reposes on the (growing) centrality of partial and weakly labeled data in statistical learning (Ratner et al., 2017). Whether this be from partial reporting in surveys, or because collecting labeled data is quite expensive, a major challenge in modern machine learning deployments and the release of statistical models is monitoring their performance. The weaker notions of predictive inference and coverage here, we might hope to build more effective and applicable guardrails and uncertainty measures for modern statistical systems, even as they are released to the world.

Appendix A. A general upper bound for the greedy approach

As we saw in section 3, reasonable conditions on label distributions guarantee that the greedy mappings $\{C_\eta^{\mathrm{gr}}\}_{\eta\in(0,1)}$ solve problem (Cond), while pathologies (as in Example 1) exist. In this section, we show that even in general cases, the sizes of the confidence sets

C^{gr} and C^{cond} cannot be too far apart. We motivate our approach by noting the similarity between problem (Cond) and the minimum set cover problem familiar in submodular optimization Vazirani (2001); Golovin et al. (2014), which we recall. Let $f : 2^{\mathcal{Y}} \to [0,1]$ be a monotone submodular coverage function, meaning that for each $A \subset B \subset \mathcal{Y}$ and $y \in \mathcal{Y} \setminus B$, f satisfies $f(A) \le f(B)$, $f(A \cup \{y\}) - f(A) \ge f(B \cup \{y\}) - f(B)$, $f(\mathcal{Y}) = 1$, and $f(\emptyset) = 0$. A solution to the minimum set cover problem is

$$C_\eta^\star \in \operatorname*{argmin}_{C \subset \mathcal{Y}} \left\{ |C| \text{ s.t. } f(C) \ge \eta \right\}. \tag{19}$$

A classical result combinatorial optimization of Wolsey (1982) bounds the size of the set that a natural greedy algorithm for problem (19) returns. To state the result, we introduce a bit of notation. For any set $C \subset \mathcal{Y}$ and $y \in \mathcal{Y}$, we define

$$\Delta(C, y) \coloneqq f(C \cup \{y\}) - f(C),$$

increase in coverage from adding y to C. At each step $j \in [K]$, the greedy algorithm chooses

$$y_j \coloneqq \operatorname*{argmax}_{y \in \mathcal{Y}} \Delta(\{y_1 \ldots, y_{j-1}\}, y),$$

and stops at the first step $j(\eta) \le K$ such that $f(\{y_1, \ldots, y_{j(\eta)}\}) \ge \eta$. For the greedy set $C^{\mathrm{gr},j} \coloneqq \{y_1, \ldots, y_j\}$, define the constant

$$K_{f,\eta} \coloneqq \min \left\{ \frac{\eta}{\eta - f(C^{\mathrm{gr},j(\eta)-1})}, \max_{\substack{y \in \mathcal{Y}, j \le j(\eta) \\ \Delta(C^{\mathrm{gr},j}, y) > 0}} \left(\frac{\Delta(\emptyset, y)}{\Delta(C^{\mathrm{gr},j}, y)} \right), \frac{\max_{y \in \mathcal{Y}} \Delta(\emptyset, y)}{\max_{y \in \mathcal{Y} \setminus C^{\mathrm{gr},j(\eta)-1}} \Delta(C^{\mathrm{gr},j(\eta)-1}, y)} \right\}.$$

We then have the following result.

Lemma 20 (Wolsey (1982), Theorem 1) *Let $f : 2^{\mathcal{Y}} \to [0,1]$ be a monotone submodular coverage function. Then*

$$|C^{\mathrm{gr},j(\eta)}| \le \left(1 + \log K_{f,\eta}\right) \cdot |C_\eta^\star|$$

Given the apparent similarity between the problems (19) and (Cond), we would like to leverage Lemma 20 to establish a similar guarantee for Alg. 2. To apply Lemma 20 to Alg. 2, we provide the natural analogous quantities, leveraging the notation in the algorithm and working conditional on $X = x$. Define $f_x(C) \coloneqq P(W \cap C \neq \emptyset \mid X = x)$, which is immediately a submodular coverage function, and for each x we have increment function $\Delta_x(C, y) = P(W \cap C = \emptyset, y \in W \mid X = x)$. Because the greedy sets $C_\eta^{\mathrm{gr}}(x, u)$ may be randomized but always satisfy $C_\eta^{\mathrm{gr}}(x, 1) \subset C_\eta^{\mathrm{gr}}(x, 0)$, we provide a slight alternative to the constant $K_{f,\eta}$, defining

$$K_{P,\eta,x} \coloneqq \min \left\{ \frac{\eta}{\eta - P(W \cap C_\eta^{\mathrm{gr}}(x,1) \neq \emptyset \mid X = x)}, \max_{\substack{y \in \mathcal{Y}, j \le j(x,\eta) \\ \Delta_x(C^{\mathrm{gr},j}(x), y) > 0}} \left(\frac{\Delta_x(\emptyset, y)}{\Delta_x(C^{\mathrm{gr},j}(x), y)} \right), \frac{\max_{y \in \mathcal{Y}} \Delta(\emptyset, y)}{\max_{y \in \mathcal{Y}} \Delta(C_\eta^{\mathrm{gr}}(x,1), y)} \right\}. \tag{20}$$

Invoking Lemma 20 and simplifying gives the following result, which bounds the size of the greedy set by a logarithmic quantity times the size of the best (deterministic) covering set.

Corollary 21 *Let $C_\eta^{\mathrm{gr}} : \mathcal{X} \times [0,1] \rightrightarrows \mathcal{Y}$ be the confidence set mapping Algorithm 2 outputs. Then for all $x \in \mathcal{X}$ and $u \in [0,1]$,*

$$|C_\eta^{\mathrm{gr}}(x,u)| \leq |C_\eta^{\mathrm{gr}}(x,0)| \leq \left(1 + \log K_{P,\eta,x}\right) \cdot \min_{C \subset \mathcal{Y}} \left\{|C| \ \textit{s.t.}\ P(W \cap C \neq \emptyset \mid x) \geq \eta\right\}.$$

We can roughly interpret the three terms inside the minimum in (20) as follows. The first term is large when the greedy algorithm nearly attains the required coverage on the iteration just before terminating, and therefore measures (in a sense) how "wasteful" the algorithm is. The second term is large when choosing a label earlier would have improved the coverage more, and so expresses a kind of regret. The third term measures how often the labels $y \in \mathcal{Y}$ co-occur in W. Though the bound is a functional of the discrete derivative $\Delta_x(C,y)$ and small when the "local" information in $\Delta_x(C,y)$ gives good indicators of globally optimal sets C, it can be hard to compute explicitly; we therefore evaluate the size of the sets that Alg. 2 generates for a few experimental examples in Section 5.

Appendix B. Proofs of mathematical results

B.1 Proofs of lower bounds on confidence set sizes

B.1.1 Proof of Theorem 3

Suppose that P is a consistent distribution on $\mathcal{X} \times \mathcal{Y} \times 2^{\mathcal{Y}}$, with marginal P_{weak} over $\mathcal{X} \times 2^{\mathcal{Y}}$, and consider a procedure $\widehat{C}_n$ offering $1-\alpha$ strong distribution-free coverage. Let $\tilde{P}$ be the distribution on $\mathcal{X} \times \mathcal{Y} \times 2^{\mathcal{Y}}$ with $\tilde{P}_{\mathrm{weak}} = P_{\mathrm{weak}}$, and where we define $\tilde{P}$ by the triple $(\tilde{X}, \tilde{Y}, \tilde{W}) \sim \tilde{P}$ according to

$$\tilde{Y} = \operatorname*{argmin}_{y \in \tilde{W}} \left\{ p_n(\tilde{X}, y) \coloneqq \mathbb{P}_{(X_i,W)_{i=1}^n \overset{\mathrm{iid}}{\sim} P_{\mathrm{weak}}} \left[y \in \widehat{C}_n(\tilde{X}) \right] \right\}.$$

Then, $\tilde{P}$ is a consistent distribution on $\mathcal{X} \times \mathcal{Y} \times 2^{\mathcal{Y}}$, which ensures that

$$\mathbb{P}_{(X_i,Y_i,W_i)_{i=1}^{n+1} \overset{\mathrm{iid}}{\sim} \tilde{P}} \left[Y_{n+1} \in \widehat{C}_n(X_{n+1}) \right] \geq 1 - \alpha.$$

By definition of $\tilde{P}$, we have

$$\mathbb{P}_{(X_i,Y_i,W_i)_{i=1}^{n+1} \overset{\mathrm{iid}}{\sim} \tilde{P}} \left[Y_{n+1} \in \widehat{C}_n(X_{n+1}) \right] = \mathbb{E}_{(X_{n+1},Y_{n+1},W_{n+1}) \sim \tilde{W}} \left[p_n(X_{n+1}, Y_{n+1}) \right],$$

the law of $\widehat{C}_n$ is identical under P or $\tilde{P}$, as it only depends on $(X_i, W_i)_{i=1}^n \overset{\mathrm{iid}}{\sim} P_{\mathrm{weak}}$.

On the other hand, we observe that when $(X_{n+1}, Y_{n+1}, W_{n+1}) \sim \tilde{P}$,

$$p_n(X_{n+1}, Y_{n+1}) = \inf_{y \in W_{n+1}} p_n(X_{n+1}, y),$$

which guarantees that

$$\begin{aligned}1-\alpha &\le \mathbb{E}_{(X_{n+1},Y_{n+1},W_{n+1})\sim\tilde{P}}\left[p_n(X_{n+1},Y_{n+1})\right] \\ &= \mathbb{E}_{(X_{n+1},Y_{n+1},W_{n+1})\sim\tilde{P}}\left[\inf_{y\in W_{n+1}} p_n(X_{n+1},y)\right] \\ &= \mathbb{E}_{(X_{n+1},W_{n+1})\sim\tilde{P}_{\text{weak}}}\left[\inf_{y\in W_{n+1}} p_n(X_{n+1},y)\right] \\ &= \mathbb{E}_{(X_{n+1},W_{n+1})\sim P_{\text{weak}}}\left[\inf_{y\in W_{n+1}} p_n(X_{n+1},y)\right].\end{aligned}$$

B.1.2 Proof of Corollary 5

Consider $(X,W)\sim P_{\text{weak}}$ independent of $(X_i,W_i)_{i\ge 1}$, and define $p(x,y)=\mathbb{P}(y\in C(x))$, recalling the definition $p_n(x,y)=\mathbb{P}(y\in\widehat{C}_n(x))$. By Jensen's inequality,

$$\begin{aligned}\mathbb{E}\left[\left|\inf_{y\in W} p(X,y)-\inf_{y\in W} p_n(X,y)\right|\right] &\le \sum_{y\in\mathcal{Y}}\mathbb{E}\left[|p(X,y)-p_n(X,y)|\right] \\ &\le \mathbb{E}\left[\sum_{y\in\mathcal{Y}}\left|1\{y\in\widehat{C}_n(X)\}-1\{y\in C(X)\}\right|\right]=\mathbb{E}\left[|\widehat{C}_n(X)\triangle C(X)|\right].\end{aligned}$$

Taking the limit as $n\to\infty$ and using that $\mathbb{E}[\inf_{y\in W} p_n(X,y)]\ge 1-\alpha$ from Theorem 3, we must also have

$$\mathbb{E}\left[\inf_{y\in W} p(X,y)\right]\ge 1-\alpha.$$

Since $\inf_{y\in W} p(X,y)\in[0,1]$, we must have $\mathbb{P}(\inf_{y\in W} p(X,y)=0)\le\alpha$, which gives the desired result: there exists $y\in W\cap\text{Det}_C(x)\setminus C(x)$ if and only if $y\in W\cap\{y\in\mathcal{Y}:p(x,y)=0\}$, i.e. if and only if $\inf_{y\in W} p(x,y)=0$.

B.2 Proofs on size set optimality in weak supervision

B.2.1 Proof of Lemma 10

Fix $\eta_0\in(0,1)$. Then $y\in C_{\eta_0}(x,u)$ implies that $s^{\text{nest}}(x,y,u)=\inf\{\eta\mid y\in C_\eta(x,u)\}\le\eta_0$ and so $s^{\text{nest}}(x,y,u)\le\eta_0$. Conversely, assume that $s^{\text{nest}}(x,y,u)\le\eta_0$. Then by definition of s^{nest}, $y\notin C_{\eta_0}(x,u)$ if and only if for all $\eta>\eta_0$, we have $y\in C_\eta(x,u)$ but $y\notin C_{\eta_0}(x,u)$, and therefore $s^{\text{nest}}(x,y,u)=\eta_0$. But of course, by continuity, $\mathbb{P}(s^{\text{nest}}(x,y,U)=\eta_0)=0$, and so

$$\mathbb{P}(s^{\text{nest}}(x,y,U)\le\eta_0 \text{ and } y\notin C_{\eta_0}(x,U))=0.$$

B.2.2 Proof of Proposition 15

The case where $W\mid X=x$ has a label-independent structure is immediate, hence we focus on proving the result when $W\mid X=x$ has a label tree-structure (8).

We prove the result by induction on the size of $\mathcal{Y}^\star$, observing that the result is immediate if $|\mathcal{Y}^\star|=1$. If $|\mathcal{Y}^\star|=K>1$, we assume that the result holds on sets with at most $K-1$ elements.

We denote P_x the law of $W \mid X = x$, by P_u the law of U and $\mathbb{P} = P_x \otimes P_u$ their joint distribution, and similarly for their expectations.

Fix $\eta \in (0,1)$, and let $C : [0,1] \rightrightarrows \mathcal{Y}^\star$ be a confidence set mapping satisfying

$$\mathbb{P}(C(U) \cap W \neq \emptyset) \geq \eta.$$

We will prove that

$$E_u|C_\eta^{\text{Greedy}}(x,U)| \leq E_u|C(U)|.$$

We use the label ranking $y_1(x), \ldots, y_K(x)$ that Alg. 2 defines, omitting x for simplicity, and consider two cases:

- **Case 1:** $P_u(y_K \in C(U)) = 0$.

  Then C provides coverage at level η using only the $K-1$ first labels, which also guarantees that $C_\eta^{\text{Greedy}}(x,u)$ only contains labels in $\{y_1, \ldots, y_{K-1}\}$ (since, in that case, $J_\eta \leq K-1$ in Alg. 2). The induction hypothesis applied to the distribution of $W \setminus \{y_K\}$ thus ensures that $E_u|C_\eta^{\text{Greedy}}(x,U)| \leq E_u|C(U)|$.

- **Case 2:** $P_u(y_K \in C(U)) > 0$.

  In that case, we will prove that either $P_u(y_j \in C(U)) = 1$ for all $j \in [K-1]$, or that we can build a new confidence set $C^{\text{final}}(U)$ such that

  $$\mathbb{P}(C^{\text{final}}(U) \cap W \neq \emptyset \mid X = x) \geq \mathbb{P}(C(U) \cap W \neq \emptyset \mid X = x),\ E_u|C^{\text{final}}(x,U)| = E_u|C(U)|,$$

  and verifies either $P_u(y_j \in C^{\text{final}}(U)) = 1$ for all $j \in [K-1]$, or $P_u(y_K \in C^{\text{final}}(U)) = 0$.

  The distribution P_x induces a tree whose leaves are the labels $y_1, \ldots, y_K$, and each inner node N (apart from the root, which is $\mathcal{Y}^\star$ itself) is a subset of $\mathcal{Y}^\star$ such that $P_x(W = N) > 0$, and two nodes N_1 and N_2 share the same parent if any subset C containing strictly either N_1 or N_2 such that $P_x(W = C) > 0$ contains $N_1 \cap N_2$. This parent is then the smallest subset N_p such that $N_1 \cap N_2 \subset N_p$ and $P_x(W = N_p) > 0$. Each parent is then the union of all its children. Figure 1 provides an example of such a tree.

  Defining $D(C) \coloneqq \{y \in \mathcal{Y}^\star \setminus \{y_K\} \mid P_u(y_j \in C(U)) < 1\} \neq \emptyset$, we consider the element $\tilde{y} \in D(C)$ sharing the lowest common ancestor with y_K in the tree. For instance, in Figure 1, if $D(C) = \{y_3, y_4\}$, then $y_D = y_3$, as their common ancestor W_0 is lower than the common ancestor of y_5 and y_4 ($\mathcal{Y}^\star$ itself).

  We then proceed to define $\tilde{C}(U)$ from $C(U)$ so that

  $$\tilde{C}(U) \setminus \{y_K, y_D\} = C(U) \setminus \{y_K, y_D\} \tag{21}$$

  and

  $$E_u|\tilde{C}(U) \cap \{y_K, y_D\}| = E_u|C(U) \cap \{y_K, y_D\}|, \tag{22}$$

  but now either

  $$P_u(y_D \in \tilde{C}(U)) = 1 \text{ or } P_u(y_K \in \tilde{C}(U)) = 0.$$

In practice, we do so by replacing y_K by y_D when $C(U) \cap \{y_K, y_D\} = \{y_K(x)\}$, or/and decreasing the probabilities that $\tilde{C}(U) \cap \{y_K, y_D\} = \{y_D, y_K\}$ and $C(U) \cap \{y_K, y_D\} = \emptyset$, in such a way that the average size does not vary, but the probability that $C(U) \cap \{y_K, y_D\} = \{y_D\}$ increases.

We then proceed to check that such a change cannot hurt our coverage, while it evidently leaves the average confidence set size unchanged (because of equations (21) and (22)).

The only way we can have $C(U) \cap W \neq \emptyset$ and $\tilde{C}(U) \cap W = \emptyset$ is if $W = \{y_K\}$. On the other hand, when $W = \{y_D\}$, we can have $C(U) \cap W = \emptyset$ but $\tilde{C}(U) \cap W \neq \emptyset$. Because of the definition of y_D with respect to y_K, any other value of W such that $C(U) \cap W \neq \emptyset$ will be such that $\tilde{C}(U) \cap W \neq \emptyset$. In particular, by independence of W and U, we have

$$\begin{aligned}
&\mathbb{P}(\tilde{C}(U) \cap W \neq \emptyset) - \mathbb{P}(C(U) \cap W \neq \emptyset) \\
&\geq P_x(W = \{y_K(x)\}) \left[P_u(y_K \in \tilde{C}(U)) - P_u(y_K \in C(U))\right] \\
&+ P_x(W = \{y_D\}) \left[P_u(y_D \in \tilde{C}(U)) - P_u(y_D \in C(U))\right] \\
&= \Big(P_u(y_D \in \tilde{C}(U)) - P_u(y_D \in C(U))\Big) \left(P_x(W = \{y_D\}) - P_x(W = \{y_K\})\right),
\end{aligned}$$

since $P_u(y_K \in \tilde{C}(U)) + P(y_D \in \tilde{C}(U)) = P_u(y_K \in C(U)) + P(y_D \in C(U))$, as the total average size does not vary.

In addition, since y_K gets selected last in Alg. 2, we know that for all $y \in \mathcal{Y}^\star$,

$$P_x\left(W = \{y_D\}\right) \geq P_x\left(W = \{y_K\}\right),$$

which achieves to prove that

$$\mathbb{P}(\tilde{C}(U) \cap W \neq \emptyset) \geq \mathbb{P}(C(U) \cap W \neq \emptyset) \geq \eta.$$

If $P_u(y_D \in \tilde{C}(U)) = 1$, then $|D(\tilde{C})| \leq |D(C)| - 1$, and we can repeat the process until we obtain a final mapping C^{final} such that either $D(C^{\text{final}}) = \emptyset$ or $\mathbb{P}(y_K(x) \in C^{\text{final}}(U)) = 0$.

In the first scenario where eventually $D(C^{\text{final}}) = \emptyset$, it means that $C^{\text{final}}(U)$ is either $\mathcal{Y}^\star$ or $\mathcal{Y}^\star \setminus \{y_K\}$, and this is immediate to check that since $\mathbb{P}(C^{\text{final}}(U) \cap W \neq \emptyset \mid X = x) \geq \eta$, we must have $P_u(y_K \in C^{\text{final}}(U)) \geq P_u(y_K \in C_\eta^{\text{Cond-Prox}}(x, U))$, which in turn ensures that

$$E_u|C_\eta^{\text{Cond-Prox}}(x, U)| \leq E_u|C^{\text{final}}(U)| = \mathbb{E}|C(U)|.$$

In the second case, since $P_U(y_K \in C^{\text{final}}(U)) = 0$, C^{final} is effectively a confidence set over strictly less than K labels, in which case we can apply the induction hypothesis to conclude that

$$E_u|C_\eta^{\text{Cond-Prox}}(x, U)| \leq E_u|C^{\text{final}}(U)| = E_u|C(U)|.$$

B.3 Proofs of algorithms validity

B.3.1 Proof of Lemma 19

We prove the result by proving that if we run Algorithm 3 with $M = K!$, defining at each step $y^m_{j,2}$ as in equation (11), then at each step $m \leq K!$ of the algorithm, $\{\mathcal{Y}^m_j\}_{j\in[m]}$ is a valid partition of $\mathcal{Y}$ that satisfies the following conditions:

1. For each $j \in [m]$, we have $\mathcal{Y}^m_j = \{y^m_j\}$ if and only if there exists no $i \in [K-1]$ such that $(i+1\ i) \circ y^m_j \in \mathcal{Y}^m_j$.
2. If $\mathcal{Y}^m_j \neq \{y^m_j\}$ then $s(x, y^m_j) \leq s(x, y^m_{j,2})$.

If the partition satisfies these two conditions at every step, then we can run the algorithm until step $m = K!$, at which point it returns a partition $\{\mathcal{Y}^{K!}_j\}^{K!}_{k=1}$ such that $s(x, y^{K!}_1) \leq \cdots \leq s(x, y^{K!}_{K!})$. Now, since we have $s^m_j = s^{K!}_j$ for every $1 \leq j \leq m$, we conclude that, at each step $m \in [K!]$, we have

$$s(x, y^m_1) \leq s(x, y^m_2) \leq \cdots \leq s(x, y^m_m) \leq \min_{y\in\mathcal{Y}\setminus\{y^m_1,\ldots,y^m_m\}} s(x, y),$$

which proves the validity of the algorithm.

This is of course true for $m = 1$, since the best configuration simply ranks $r_k(x)$ in decreasing order.

1. By definition of $y^m_{\mathrm{ind}(m),2}$ and $y_{\mathrm{ind}(m)}$, $\{\mathcal{Y}^{m+1}_j\}^{m+1}_{j=1}$ is a valid partition of $\mathcal{Y}$ such that each $y^{m+1}_j \in \mathcal{Y}^{m+1}_j$, if $\{\mathcal{Y}^m_j\}$ is itself a valid partition (and $m < K!$), so long as we can prove that if $\mathcal{Y}^m_j \neq \{y^m_j\}$, then there must exist $\alpha \in [K-1]$ such that $(\alpha+1\ \alpha) \circ y^m_j \in \mathcal{Y}^m_j$ (i.e the algorithm does not get stuck and terminates). But this is immediate as, if for all $\alpha \in [K-1]$, we have $(\alpha+1\ \alpha) \circ y^m_j \notin \mathcal{Y}^m_j$, then it must be by construction that

$$\mathcal{Y}^m_j \subset \bigcap_{\alpha\in[K-1]} \{y \in \mathcal{Y} \mid y^{-1}(y^m_j(\alpha)) < y^{-1}(y^m_j(\alpha+1))\} = \{y^m_j\}.$$

   Therefore, at each step $m \leq K!$ of the algorithm, $\{\mathcal{Y}^m_j\}_{j\in[m]}$ is a valid partition of $\mathcal{Y}$, and the algorithm terminates.

2. On the other hand, it requires more care to justify why, if we set, for all $j \leq m$,

$$y^m_{j,2} \coloneqq \operatorname*{argmin}_{y\in\mathcal{Y}^m_j} \{s(x, y) \mid \exists \alpha \in [K-1],\ y = (\alpha+1\ \alpha) \circ y^m_j\}$$

   then we should always have

$$s(x, y^m_j) \leq s(x, y^m_{j,2}) \tag{23}$$

   for all $j \in [m]$, i.e. why any permutation of the form $(\alpha\ \alpha+1) \circ y^m_j$ that belongs to $\mathcal{Y}^m_j$ cannot strictly decrease the score $s(x, y)$.

Equation (23) actually results from a crucial property of the score function (10), which ensures that if $s(x,y) < s(x,(\alpha\ \alpha+1)\circ y)$, then it must hold that

$$r_{y(\alpha)}(x) < r_{y(\alpha+1)}(x),$$

i.e. the elements $y(\alpha)$ and $y(\alpha+1)$ were originally in the wrong order in y.

But, since we start the partition process with y_1^1 such that $r_{y_1^1(1)}(x) \geq \cdots \geq r_{y_1^1(k)}(x)$, i.e with all elements in the correct order, it is straightforward to check that at any time m, there cannot exist a permutation of the form $(\alpha\ \alpha+1)\circ y_j^m$ that also belongs to $\mathcal{Y}_j^m$ such that

$$r_{y_j^m(\alpha)}(x) < r_{y_j^m(\alpha+1)}(x) :$$

if that were the case, then there would exist $l \leq j$ such that $y_j^m(\alpha)$ and $y_j^m(\alpha+1)$ were the elements exchanged at time l when creating the partition $\{\mathcal{Y}_i^{l+1}\}_{i=1}^{l+1}$. In turn, since $y_j^m \in \mathcal{Y}_j^m$, this would guarantee that

$$\mathcal{Y}_j^m \subset \{y \in \mathcal{Y} \mid y^{-1}(y_j^m(\alpha)) < y^{-1}(y_j^m(\alpha+1))\},$$

and thus that $(\alpha\ \alpha+1)\circ y_j^m \notin \mathcal{Y}_j^m$. This guarantees that any configuration $(\alpha\ \alpha+1)\circ y_j^m \in \mathcal{Y}_j^m$ satisfies

$$s(x,(\alpha\ \alpha+1)\circ y_j^m) \geq s(x,y_j^m),$$

and thus that either $\mathcal{Y}_j^m = \{y_j^m\}$, or

$$s(x,y_{j,2}^m) \geq s(x,y_j),$$

which concludes the proof.

Appendix C. Further experiments

C.1 Structured prediction example (Perfect matching scores and weak supervision)

A matching task consists of trying to optimally pair elements of a bipartite graph given a feature vector $x \in \mathcal{X}$ (for example, one may wish to identify paired amino acids in protein folding Taskar et al. (2003)). We assume there exists a bipartite graph G with disjoint sets U and V of $K \geq 1$ nodes; each label Y is then a perfect matching between U and V, i.e., a bijection $Y \in \mathcal{Y} = \mathfrak{S}(U,V)$. General supervised approaches for perfect matching problems, such as structured Support Vector Machines Tsochantaridis et al. (2004) or Adversarial Bipartite Matching Fathony et al. (2018), generally learn pairwise score functions $\varphi_{u,v} : \mathcal{X} \to \mathbb{R}$ for all $(u,v) \in U \times V$, which measure the cost of adding the edge $e \coloneqq (u,v)$ for a feature vector $x \in \mathcal{X}$, and then output a prediction

$$y^\star(x) \coloneqq \left\{ \operatorname*{argmin}_{y \in \mathfrak{S}(U,V)} \sum_{u \in U} \varphi_{u,y(u)}(x) = \sum_{u \in U, v \in V} 1\{v = y(u)\}\, \varphi_{u,v}(x) \right\},$$

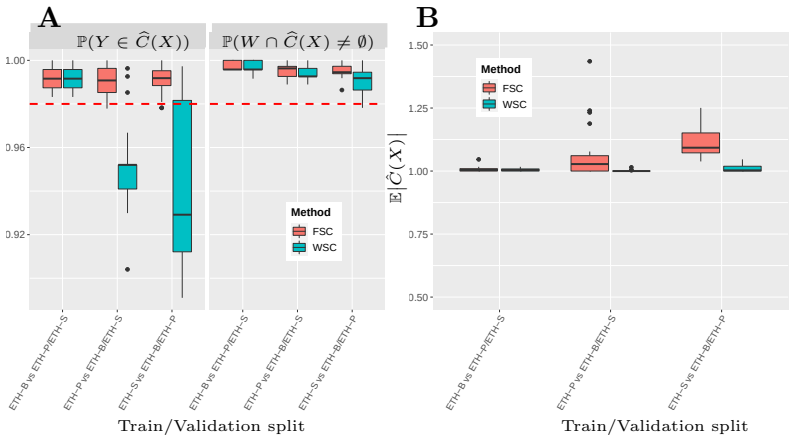

Figure 6. Results for the video tracking matching dataset MOT2015 Leal-Taixé et al. (2015), over $N_{\text{trials}} = 20$ trials. We use one video sequence for training (ETH-B, ETH-P or ETH-S) and the two others for calibration and testing. A: Strong (1) and weak (3) coverage. B: Average confidence set size for fully supervised (FSC) and weakly supervised conformal (WSC) methods.

an instance of minimum cost perfect matching solvable in time $\mathcal{O}(K^3)$ with the Hungarian algorithm. To efficiently adapt this approach in the context of Alg. 1, we assume we have trained a set of pairwise score functions $\{\varphi_{u,v} \mid (u,v) \in U \times V\}$, (e.g. using supervised training data) and wish to conformalize with partially supervised data, using the following score function

$$s^{\text{Matching}}(x,y) \coloneqq \sum_{u \in U, v \in V} 1\{v = y(u)\}\, \varphi_{u,v}(x,y). \tag{24}$$

In a matching problem, weak supervision can arise under the form of a partial matching between subsets $U_i \subset U$ and $V_i \subset V$ of the nodes, which we write $Y_i^{\text{weak}} \in \mathfrak{S}(U_i, V_i)$: each $u \in U_i$ has a matching element $Y_i^{\text{weak}}(u) = Y_i(u) \in V_i$. Computing the minimum partial scores (5) in Alg. 1 is then computationally efficient, as it reduces to yet another minimum cost perfect matching problem:

$$S_i \coloneqq \sum_{u \in U_i} \varphi_{u, Y_i^{\text{weak}}(u)}(x) + \min_{\tilde{y} \in \mathfrak{S}(U \setminus U_i, V \setminus V_i)} \left\{ \sum_{\substack{u \in U \setminus U_i \\ v \in V \setminus V_i}} 1\{v = \tilde{y}(u)\}\, \varphi_{u,v}(x) \right\}.$$

In the matching case, Alg. 3 is equivalent to finding the M-best minimal weight perfect matching in a bipartite graph, which Chegireddy and Hamacher (1987) efficiently solve. In the context of Alg. 3, their approach iteratively chooses an edge $e_m \in y^m_{\text{ind}(m)} \setminus y^m_{\text{ind}(m),2}$, then partitions the set of matchings $\mathcal{M} \in \mathcal{Y}^m_{\text{ind}(m)}$ depending on whether $e_m \in \mathcal{M}$ or not. The computation of each second-best configuration then amounts to solving at most K different perfect matching problems, resulting in an overall $\mathcal{O}(MK^4)$ cost of the procedure.

C.2 Pedestrian tracking with partial matching information

We now apply our weakly supervised conformal methods to a bipartite matching problem. A common objective in computer vision, relevant for instance for self-driving cars, is to

track people's trajectory throughout different time frames. Since we can leverage powerful algorithms Redmon et al. (2016) to individually detect objects in every single frame, the problem that we study here is actually a matching task where the goal is to match two sets of people appearing in two separate frames: this is an instance of a maximal matching problem.

Weak supervision with partial matchings In this context, we expect partial supervision to come under the form of a partial matching: some people, e.g. people that are easier to track between two consecutive frames because they are in the foreground, already have their match in the second frame, when others, perhaps more difficult to track, are still waiting for a potential match. Given these partially labeled instances, the goal then becomes to return confidence sets of matchings that guarantee $1-\alpha$ coverage: to provide valid weak coverage (3), we wish to include a configuration that contains at least all the partial matches.

Predicting trajectories in the MOT2D15 data set Leal-Taixé et al. (2015) We experiment using the MOT2D15 pedestrian video tracking dataset Leal-Taixé et al. (2015). It is a public benchmark data set that contains short street videos with pedestrians, and for which the goal is to track each of them while they appear in the frame. Specifically, each frame already has a set of bounding boxes corresponding to each individual present in that frame, and we aim at matching boxes representing the same person between two consecutive frames. Since an individual can enter or exit the frame between two consecutive images, we need to account for potentially unmatched boxes, which we do by including "virtual" nodes in the bipartite graph, similarly to previous approaches Kim et al. (2013); Fathony et al. (2018).

We use the same feature representation of Kim et al. (2013) and Fathony et al. (2018): given a pair $x \coloneqq (x_1, x_2)$ of two consecutive images, and two bounding boxes $u \subset x_1$, $v \subset x_2$, we compute a $d = 46$ dimensional vector $\phi(x_1, x_2, u, v)$ that summarizes key features (e.g. position of the bounding box, color distribution) allowing to determine whether u and v represent the same person. We then train our model using a structured S-SVM approach Tsochantaridis et al. (2004), following the approach of Kim et al. (2013). Such model outputs a pairwise score function $s^{\mathrm{PW}} : (x, u, v) \mapsto \theta^T \phi(x_1, x_2, u, v)$ for some vector $\theta \in \mathbb{R}^d$ where the feature vector $x = (x_1, x_2) \in \mathcal{X}$ consists of two consecutive frames, and (u, v) are two potential bounding boxes (one in each image).

Experimental set-up and partial labels We use the data set MOT2D15 as follows. For each of the ETH-Bahnhof, ETH-Pedcross2 and ETH-Sunnyday video sequences, which contain respectively 1000, 837, and 354 consecutive images, we select one of them for training, one of them for calibration and one of them for actual testing, using $\alpha = 0.02$ for conformalization purposes. We further introduce weak supervision by assuming that for each pair of images, we observe a partial matching, precisely that among the K_i paired individuals, a user provided us feedback on $K_i^{\mathrm{partial}} \overset{\mathrm{iid}}{\sim} 1 + \mathrm{Poi}(0.5)$ matches.

For both our FSC and WSC methods, we use a translated version of the score s^{Matching} (24) with pairwise functions $\{s_{u,v} : x \mapsto s^{\mathrm{PW}}(x, u, v)\}_{u,v}$: for each instance $(x, y) \in \mathcal{X} \times \mathcal{Y}$, we

use the score

$$\tilde{s}^{\text{Matching}}(x, y) \coloneqq s^{\text{Matching}}(x, y) - \min_{\tilde{y} \in \mathcal{Y}} s^{\text{Matching}}(x, \tilde{y})$$

This operation simply ensures that $\min_{y \in \mathcal{Y}} \tilde{s}^{\text{Matching}}(x, y) = 0$ for every $x \in \mathcal{X}$ and thus does not change the ordering of configurations, nor the score difference between two configurations. We simply use it to make sure to place all the instances X_i on the same scale when applying Algorithm 1 in the sense that they all have the same best achievable score. In particular, in the noiseless case where the true label Y_i is always the minimizer of $y \mapsto s^{\text{Matching}}(X_i, y)$, it guarantees that $\widehat{C}(X)$ eventually contains a single configuration (as we should, since the score function in this case outputs perfect predictions).

Experimental results This specific problem is actually low-noise, as it is possible to achieve a very high accuracy with the S-SVM approach, which is not so surprising as we assume perfect detection of every person thanks the bounding boxes. As a result, we can expect the FSC and WSC methods to output very similar confidence sets, as the configuration minimizing the score is often the true label itself. This is precisely what we observe in Figure 6, where both methods are actually indistinguishable and where, even with a very high confidence $1 - \alpha. = 0.98$, both the FSC and WSC methods are able to return confidence sets with a single configuration in average. We only notice a slight difference between both methods when training on the ETH-Sunnyday sequence, which contains fewer images, and hence produces slightly worse score functions.

C.3 Prediction intervals for weakly supervised regression

As we mentioned earlier, much of our development thus far goes beyond (finite) spaces with combinatorial structure, as we saw in the last few examples. Therefore, we finish with a regression problem. We consider predicting the fraction of votes in each United States county for the Democratic Party candidate in the 2016 United States presidential election, using demographic (census) data as covariates and the results of past elections. It is common during elections for forecasters to build predictive models from both census and historical election data, as well as current polling data. We view the historical data as strong supervision (it tells us exactly how many people voted for each candidate), and the polling data as weak supervision (as polls always come with a margin of error). Our goal here is to fit a regression model to the strongly supervised past election data, and then form prediction intervals for the fraction of people in each county who voted Democrat by leveraging the weakly supervised polling data. We hope by combining both we obtain valid intervals narrower than the polling margins of error.

Our data comes from the 2013–2017 American Community Survey 5-Year Estimates, which is a longitudinal survey that records demographic information about each of the 3220 United States counties. We use 34 of the available demographic features. The response is simply the fraction of people in each county who voted Democratic (and is publicly available). Thus our sample size is 3220, and the number of dimensions is $d = 34$.

We construct a stylized version of the above situation in the following way. First, we split the data set into thirds: 33% of the counties (and their associated fractions of Democratic voters) go into the training set, 33% go into the calibration set, and the rest go into the test

set; as our splits are random, they are exchangeable. Then, we fit a Beta regression model to the strongly supervised training data. To simulate the availability of weakly supervised polling data, we replace each calibration set response $Y_i \in [0,1]$ with a weak response $W_i \subset [0,1]$, $i = 1, \ldots, n$, by forming

$$W_i = [Y_i - Z_i, Y_i + Z_i], \quad Z_i \sim \mathsf{N}(\mu, 0.0001), \quad i = 1, \ldots, n, \tag{25}$$

for various values of $\mu \in \{.01, .05, .1, .15, .2\}$, so that W_i captures fluctuations around Y_i that are roughly $\pm\mu$. We conformalize by running Alg. 1 with the absolute error scoring function,

$$s(x, y) = |\hat{y}(x) - y|,$$

where $\hat{y}(x) \in [0,1]$ denotes the Beta regression model's prediction for the point $x \in \mathbb{R}^d$. Here, conformalization boils down to solving a simple linear program, i.e.,

$$s(X_i, Y_i) = \min_{\gamma \in \mathbb{R}} \ \{|\hat{y}(X_i) - \gamma| \ \mid \ Y_i - Z_i \le \gamma, \ \gamma \le Y_i + Z_i\}, \quad i = 1, \ldots, n.$$

Finally, we evaluate both strong (1) and weak (3) coverage on the test set (we apply the same transformation as in (25) to generate the weak labels for the test set). We compute the two types of coverage, as well as the lengths of the prediction intervals, by repeating this process 20 times. We set the miscoverage level $\alpha = .05$.

Similar to our other experiments, we find that Alg. 1 achieves weak coverage at the nominal level .95, for all values of μ (governing the amount of weak supervision), shown in the middle panel of Figure 7. On the other hand, we expect the strong coverage to be much lower. The left panel of Figure 7 shows the strong coverage for Alg. 1 in teal and standard conformal inference in pink, and we can see that this is indeed the case. However, in return, we expect the prediction intervals that Alg. 1 generates to be smaller than those coming from standard conformal inference. The right panel of Figure 7 shows that this is also the case: in particular, when $\mu \ge .1$, the average length of Alg. 1's intervals is at least three times smaller than standard conformal's, and half the length of the average weakly supervised interval W_i from (25) ($\approx .2$). We can also see from these figures, as in our other experiments, that Alg. 1's strong coverage degrades as μ grows, whereas its weak coverage improves and the length of its prediction intervals shrinks.

We view these results from a slightly different perspective in Figures 8 and 9. In Figure 8, we show the true fraction of people in each county from the test set that voted Democratic. In Figure 9, we show the lower and upper endpoints of Alg. 1's prediction intervals, for a randomly chosen repetition with $\mu = .05$. In these two figures, we color the counties with strong (predicted) Democratic majorities blue, and those with strong (predicted) Republican majorities red. By comparing the colors, we can see that the prediction intervals only sometimes contain the true response, which is expected. Finally, we note that the colors of the lower and upper endpoints in Figure 9 are similar, because the length of the prediction intervals is usually small.

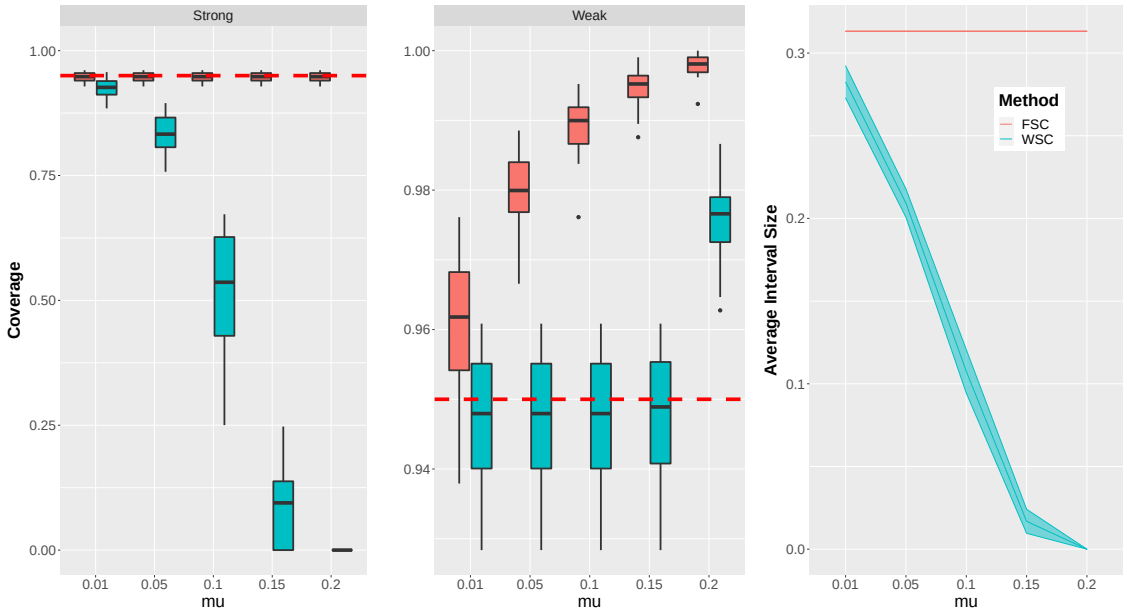

Figure 7. Results for the regression problem with the election data, over 20 trials. The left panel shows the strong coverage (1), and the middle panel shows the weak coverage (3). The dashed red line indicates the nominal coverage level, $1 - \alpha = .95$. The right panel shows the prediction interval lengths. In these plots, we show Alg. 1, denoted "WSC", in teal. We show standard conformal inference, denoted "FSC", in pink.

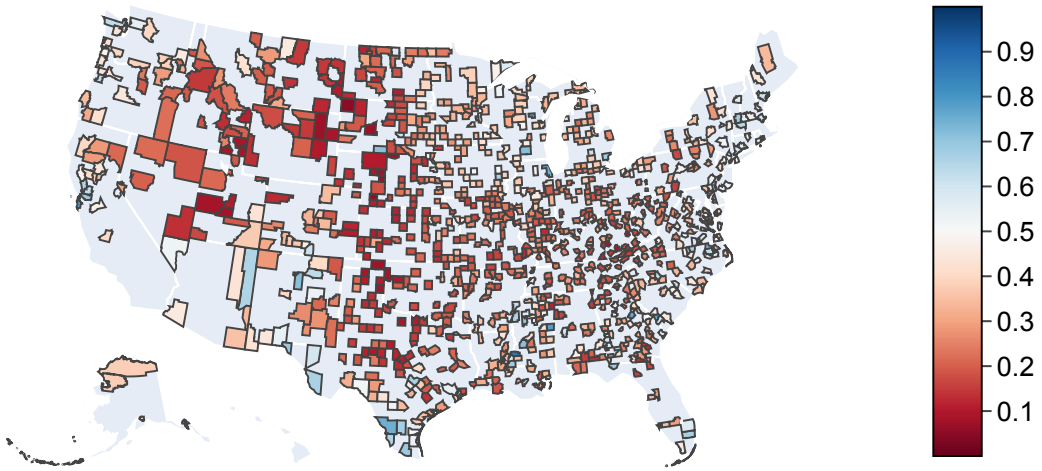

Figure 8. Map of United States counties. We color each county according to the actual fraction votes for the Democratic candidate in the 2016 United States presidential election. We color counties with strong Democratic majorities blue, and those with strong Republican majorities red. We color the counties from the training and calibration sets gray.

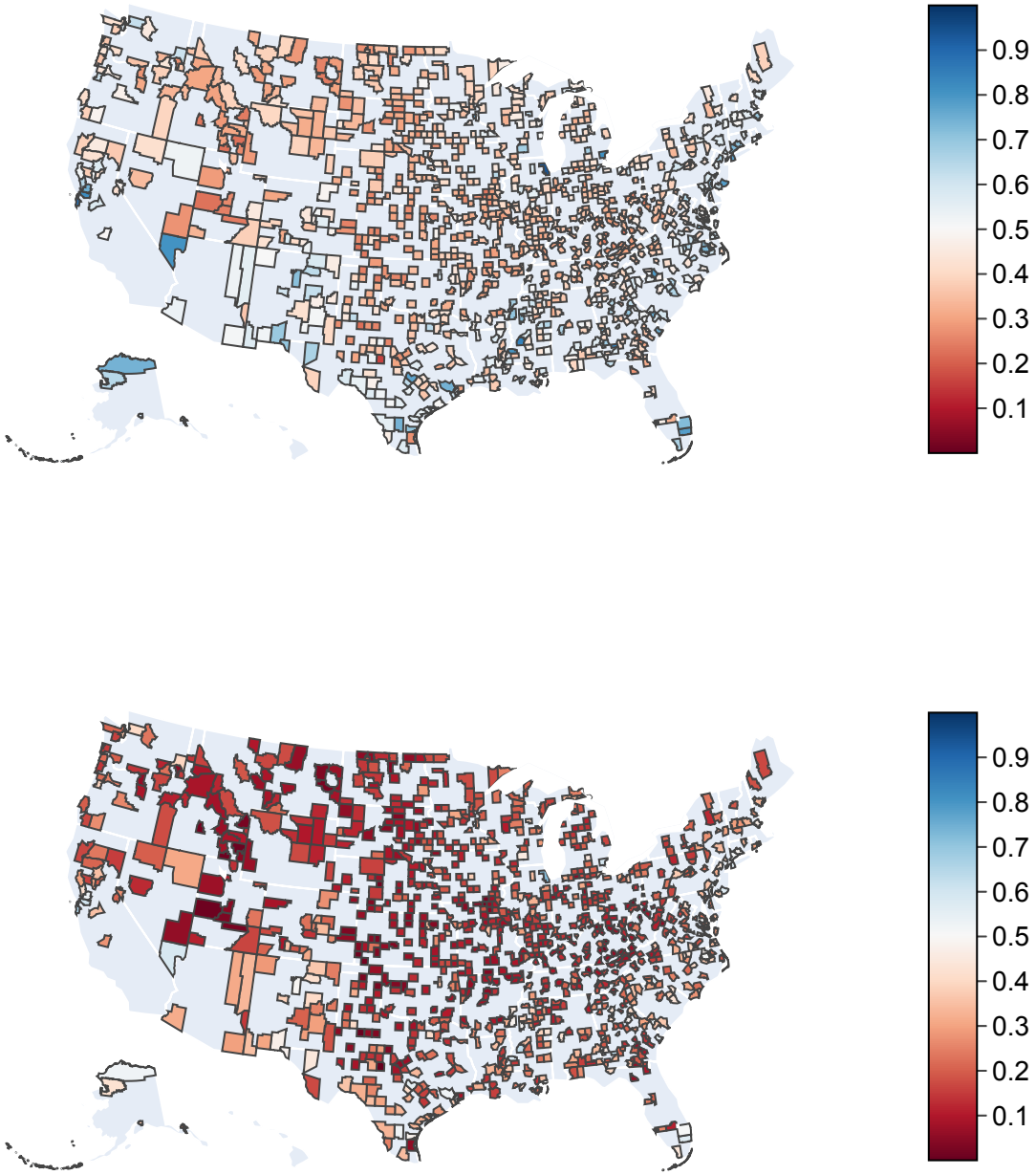

Figure 9. Map of United States counties. We color each county according to the value of the upper (top panel) and lower (bottom panel) endpoints of the confidence interval that Alg. 1 returns, when $\mu = .05$. We color counties with values close to 1 blue, and those with values close to 0 red. We color the counties from the training and calibration sets gray.

References

Nir Ailon, Moses Charikar, and Alantha Newman. Aggregating inconsistent information: Ranking and clustering. *Journal of the Association for Computing Machinery*, 55(5), 2008.

Anastasios Angelopoulos, Stephen Bates, Jitendra Malik, and Michael I. Jordan. Uncertainty sets for image classifiers using conformal prediction. *arXiv:2009.14193 [cs.CV]*, 2020.

Rina Foygel Barber, Emmanuel J. Candès, Aaditya Ramdas, and Ryan J. Tibshirani. The limits of distribution-free conditional predictive inference. *Information and Inference*, 10 (2):455–482, 2021.

Stephen Bates, Anastasios Angelopoulos, Lihua Lei, Jitendra Malik, and Michael I. Jordan. Distribution-free, risk-controlling prediction sets. *Journal of the Association for Computing Machinery*, 68(6):43:1–43:34, 2021.

Matthew R Boutell, Jiebo Luo, Xipeng Shen, and Christopher M Brown. Learning multi-label scene classification. *Pattern Recognition*, 37(9):1757–1771, 2004.

Vivien Cabannes, Alessandro Rudi, and Francis Bach. Structured prediction with partial labelling through the infimum loss. In *Proceedings of the 37th International Conference on Machine Learning*, 2020.

Zhe Cao, Tao Qin, Tie-Yan Liu, Ming-Feng Tsai, and Hang Li. Learning to rank: from pairwise approach to listwise approach. In *Proceedings of the 24th International Conference on Machine Learning*, pages 129–136, 2007.

Maxime Cauchois, Suyash Gupta, Alnur Ali, and John Duchi. Robust validation: Confident predictions even when distributions shift. *arXiv:2008.04267 [stat.ML]*, 2020.

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

Maxime Cauchois, Suyash Gupta, Alnur Ali, and John Duchi. Predictive inference with weak supervision. *arXiv:2201.08315 [stat.ML]*, 2022.

Chandra R Chegireddy and Horst W Hamacher. Algorithms for finding k-best perfect matchings. *Discrete Applied Mathematics*, 18(2):155–165, 1987.

Jesús Cid-Sueiro, Darío García-García, and Raúl Santos-Rodríguez. Consistency of losses for learning from weak labels. In *Proceedings of the European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 197–210, 2014.

Timothee Cour, Ben Sapp, and Ben Taskar. Learning from partial labels. *Journal of Machine Learning Research*, 12:1501—-1536, 2011.

J. Deng, W. Dong, R. Socher, L. Li, K. Li, and L. Fei-Fei. ImageNet: a large-scale hierarchical image database. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009.

David L. Donoho. 50 years of data science. *Journal of Computational and Graphical Statistics*, 26(4):745–766, 2017.

John C. Duchi, Lester Mackey, and Michael I. Jordan. The asymptotics of ranking algorithms. *Annals of Statistics*, 41(5):2292–2323, 2013.

John C. Duchi, Suyash Gupta, Kuanhao Jiang, and Pragya Sur. Predictive inference in multi-environment scenarios. *arXiv:2403.16336 [stat.ML]*, 2024.

Bat-Sheva Einbinder, Yaniv Romano, Matteo Sesia, and Yanfei Zhou. Training uncertainty-aware classifiers with conformalized deep learning. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems*, volume 35, pages 22380–22395. Curran Associates, Inc., 2022. URL `https://proceedings.neurips.cc/paper_files/paper/2022/file/8c96b559340daa7bb29f56ccfbbc9c2f-Paper-Conference.pdf`.

A. Elisseeff and J. Weston. A kernel method for multi-labeled classification. In *Advances in Neural Information Processing Systems 14*, 2001.

Rizal Fathony, Sima Behpour, Xinhua Zhang, and Brian Ziebart. Efficient and consistent adversarial bipartite matching. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80, pages 1457–1466, 2018.

Eraldo R. Fernandes and Ulf Brefeld. Learning from partially annotated sequences. In *Proceedings of the European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 407–422, Berlin, Heidelberg, 2011. Springer Berlin Heidelberg.

Adam Fisch, Tal Schuster, Tommi Jaakkola, and Regina Barzilay. Efficient conformal prediction via cascaded inference with expanded admission. In *Proceedings of the Ninth International Conference on Learning Representations*, 2021.

Y. Freund, R. Iyer, R. E. Schapire, and Y. Singer. Efficient boosting algorithms for combining preferences. *Journal of Machine Learning Research*, 4:933–969, 2003.

Daniel Golovin, Andreas Krause, and Matthew Streeter. Online submodular maximization under a matroid constraint with application to learning assignments. *arXiv:1407.1082 [cs.LG]*, 2014.

Chirag Gupta, Arun K. Kuchibhotla, and Aaditya K. Ramdas. Nested conformal prediction and quantile out-of-bag ensemble methods. *Pattern Recognition*, 127, 2022. Special Issue on Conformal and Probabilistic Prediction with Applications.

Suyash Gupta. *Reliability and stability in statistical and machine learning problems*. PhD thesis, Stanford University, 2022.

Suyash Gupta and Dominik Rothenhäusler. The s-value: evaluating stability with respect to distributional shifts. *arXiv:2105.03067 [stat.ME]*, 2021.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016.

Yotam Hechtlinger, Barnabás Póczos, and Larry Wasserman. Cautious deep learning. *arXiv:1805.09460 [stat.ML]*, 2019.

Eyke Hüllermeier, Johannes Fürnkranz, Weiwei Cheng, and Klaus Brinker. Label ranking by learning pairwise preferences. *Artificial Intelligence*, 172(16—17):1897—-1916, November 2008.

John G. Kemeny. Mathematics without numbers. *Daedalus*, 88(4):577–591, 1959.

M. G. Kendall. A new measure of rank correlation. *Biometrika*, 30(1/2):81–93, 1938.

Suna Kim, Suha Kwak, Jan Feyereisl, and Bohyung Han. Online multi-target tracking by large margin structured learning. In *Proceedings of the Asian Conference on Computer Vision*, pages 98–111, 2013.

Anna Korba, Alexandre Garcia, and Florence d'Alché Buc. A structured prediction approach for label ranking. In *Advances in Neural Information Processing Systems 31*, pages 8994–9004, 2018.

L. Leal-Taixé, A. Milan, I. Reid, S. Roth, and K. Schindler. MOTChallenge 2015: Towards a benchmark for multi-target tracking. *arXiv:1504.01942 [cs.CV]*, 2015.

Jing Lei. Classification with confidence. *Biometrika*, 101(4):755–769, 2014.

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

Jing Lei, Max G'Sell, Alessandro Rinaldo, Ryan J Tibshirani, and Larry Wasserman. Distribution-free predictive inference for regression. *Journal of the American Statistical Association*, 113(523):1094–1111, 2018.

Stephen Mayhew, Snigdha Chaturvedi, Chen-Tse Tsai, and Dan Roth. Named entity recognition with partially annotated training data. In *Proceedings of the 23rd Conference on Computational Natural Language Learning*, pages 645–655, 01 2019.

Sahand Negahban, Sewoong Oh, and Devavrat Shah. Iterative ranking from pair-wise comparisons. *Operations Research*, 65(1):266–287, 2016.

Nam Nguyen and Rich Caruana. Classification with partial labels. In *Proceedings of the 14th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*, pages 551–559, 2008.

G. Papandreou, L. Chen, K. P. Murphy, and A. L. Yuille. Weakly-and semi-supervised learning of a deep convolutional network for semantic image segmentation. In *Proceedings of the International Conference on Computer Vision*, pages 1742–1750, 2015.

Tao Qin and Tie-Yan Liu. Introducing LETOR 4.0 datasets. *arXiv:1306.2597 [cs.IR]*, 2013.

Alexander Ratner, Stephen H. Bach, Henry Ehrenberg, Jason Fries, Sen Wu, and Christopher Ré. Snorkel: rapid training data creation with weak supervision. *Proceedings of the VLDB Endowment*, 11(3):269–282, 2017.

Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the 26th IEEE Conference on Computer Vision and Pattern Recognition*, pages 779–788, 2016.

Yaniv Romano, Rina Barber, Chiara Sabatti, and Emmanuel J. Candes. With malice towards none: Assessing uncertainty via equalized coverage. *arXiv:1908.05428 [stat.ME]*, 2019a.

Yaniv Romano, Evan Patterson, and Emmanuel J. Candès. Conformalized quantile regression. In *Advances in Neural Information Processing Systems 32*, 2019b.

Yaniv Romano, Matteo Sesia, and Emmanuel J. Candès. Classification with valid and adaptive coverage. In *Advances in Neural Information Processing Systems 33*, 2020.

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

D. Sculley, Gary Holt, Daniel Golovin, Eugene Davydov, Todd Phillips, Dietmar Ebner, Vinay Chaudhary, Michael Young, Jean-Francois Crespo, and Dan Dennison. Hidden technical debt in machine learning systems. In *Advances in Neural Information Processing Systems 28*, 2015.

B. Taskar, C. Guestrin, and D. Koller. Max-margin markov networks. In *Advances in Neural Information Processing Systems 16*, 2003.

Ben Taskar. *Learning Structured Prediction Models: A Large Margin Approach*. PhD thesis, Stanford University, 2005.

Ryan J. Tibshirani, Rina Foygel Barber, Emmanuel J. Candès, and Aaditya Ramdas. Conformal prediction under covariate shift. In *Advances in Neural Information Processing Systems 32*, 2019.

Bill Triggs and Jakob Verbeek. Scene segmentation with crfs learned from partially labeled images. In *Advances in Neural Information Processing Systems 21*, volume 20, pages 1553–1560, 2008.

I. Tsochantaridis, T. Hofmann, T. Joachims, and Y. Altun. Support vector machine learning for interdependent and structured output spaces. In *Proceedings of the Twenty-First International Conference on Machine Learning*, 2004.

Brendan van Rooyen and Robert C. Williamson. A theory of learning with corrupted labels. *Journal of Machine Learning Research*, 18(228):1–50, 2018.

Anke van Zuylen, Rajneesh Hegde, Kamal Jain, and David P. Williamson. Deterministic pivoting algorithms for constrained ranking and clustering problems. In *Proceedings of the Eighteenth ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 405–414. Society for Industrial and Applied Mathematics, 2007.

Vijay V. Vazirani. *Approximation Algorithms*. Springer, 2001.

Vladimir Vovk, Alexander Grammerman, and Glenn Shafer. *Algorithmic Learning in a Random World*. Springer, 2005.

Laurence A. Wolsey. An analysis of the greedy algorithm for the submodular set covering problem. *Combinatorica*, 2:385–393, 1982.

Hsiang-Fu Yu, Prateek Jain, Purushottam Kar, and Inderjit S. Dhillon. Large-scale multi-label learning with missing labels. In *Proceedings of the 31st International Conference on Machine Learning*, volume 32, pages 593—601, 2014.

Ce Zhang, Chris Ré, Michael Cafarella, Christopher De Sa, Alex Ratner, Jaeho Shin, Feiran Wang, and Sen Wu. DeepDive: Declarative knowledge base construction. *Communications of the ACM*, 60(5):93–102, 2017.