Warm-starting Contextual Bandits: Robustly Combining Supervised and Bandit Feedback

Chicheng Zhang [1] **Alekh Agarwal** [1] **Hal Daumé III** [1 2] **John Langford** [1] **Sahand N Negahban** [3]

Abstract

We investigate the feasibility of learning from both fully-labeled supervised data and contextual bandit data. We specifically consider settings in which the underlying learning signal may be different between these two data sources. Theoretically, we state and prove no-regret algorithms for learning that is robust to divergences between the two sources. Empirically, we evaluate some of these algorithms on a large selection of datasets, showing that our approaches are feasible, and helpful in practice.

1. Introduction

In many real-world settings, a system must learn from multiple types of feedback; we consider the specific setting of learning jointly from fully labeled "supervised" examples and from online feedback "contextual bandit" (abbrev. CB) examples. For instance, in a system that chooses personalized content to display on a webpage, an expert may be able to provide an initial set of fully labeled examples to get a system started, but, after deployment, the system can only measure its performance (e.g., dwell time) on the content it displays and not other (counterfactual) options. In an automated translation system, professional translators can provide initial translations to seed a system, but the system may be able to further improve its performance based on, e.g., user satisfaction measures (Sokolov et al., 2015; Nguyen et al., 2017).

In both these settings (content display and translation), we desire an approach that is able to use the fully supervised expert data to "warm-start" a system, which later learns from CB feedback (Auer et al., 2002b; Langford & Zhang, 2007; Chu et al., 2011; Dudik et al., 2011; Agrawal & Goyal, 2013; Agarwal et al., 2014) Doing so has the added advantage of ensuring that such a system does not need to suffer too much error in an initial exploration phase, which may be necessary in user-facing systems or in error- or safety-critical settings (Tewari & Murphy, 2017). However, it is generally unreasonable to assume that the expert supervision and the CB feedback in such settings are perfectly aligned: the "best" decision according to an expert may not necessarily match a user's choice. We need algorithms that operate well even in the case of unknown degrees of misalignment; we introduce a hypothesis class-specific notion of cost similarity used in our analysis, but not our algorithms (§2). We also highlight how simple strategies for combining the two sources without robustness to misalignment can perform significantly worse than learning from the ground truth source alone (§2.1).

Furthermore, different applications can differ in terms of which source—supervised or CB—is considered "ground truth". For example, while the CB feedback from users is the better signal about their preferences in content personalization (§ 3), the expert translations provide the ground truth in the translation setting for which user satisfaction is an imperfect proxy (§4). We develop algorithms for *both* settings, which effectively "search" for a good balance between fitting the CB feedback and supervised labels. In both cases, we provide regret bounds showing the value of the complementary data sources, dependent on their cost discrepancy and respective sample sizes. Importantly, our theory shows that our methods perform close to an oracle that knows the similarity of the two sources beforehand and uses it to optimally weight their examples, with a small additional penalty from searching for this weighting.

Empirically, we perform experiments based on fully-labeled examples from which CB feedback is simulated. We focus on the setting when CB data is ground truth and the supervised warm-start might have differing levels of bias. In an experimental study over hundreds of datasets (§ 5) we demonstrate the efficacy of our algorithm. As a snapshot, we show an empirical cumulative distribution function (CDF) across a number of experimental conditions, where each (x, y) value on the curve indicates that there is a y fraction of experimental conditions where the normalized error[1] of a method is below x. The plot aggregates across settings where the CB and supervised signals are perfectly

[1]Microsoft Research [2]University of Maryland [3]Yale University. Correspondence to: Chicheng Zhang <Chicheng.Zhang@microsoft.com>.

[1]See §5 for a formal definition of normalized error.

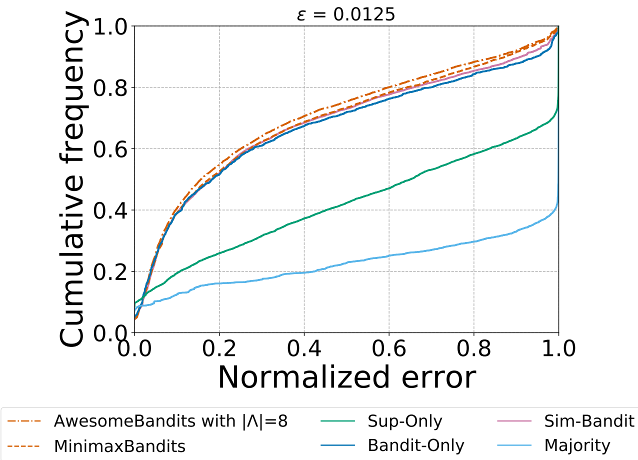


Figure 1: Empirical CDF of the performance of different methods across a number of datasets and experimental conditions (See §5 for detailed descriptions of all algorithms, settings, and aggregation method). Our method AWESOMEBANDITS is evaluated with a parameter $|\Lambda| = 8$; MINIMAXBANDITS and SIM-BANDIT are baselines also leveraging the warm-start; SUP-ONLY, MAJORITY[1] and BANDIT-ONLY learn using only the supervised and CB sources respectively. All CB methods use ϵ-greedy with $\epsilon = 0.0125$.

aligned as well as where they are not. Overall, our method, named AWESOMEBANDITS, outperforms all baselines in this aggregated summary, in particular beating the two baselines that leverage both CB and supervised sources. More detailed results are presented in §5.

Relation to prior work. A theoretical study of domain adaptation (Ben-David et al., 2010; Mansour et al., 2009) and learning from multiple sources (Crammer et al., 2008) are the most relevant prior works. In these works, all data sources provide the same supervised feedback rather than the supervised/CB modality we investigate here, with the two sources having drastically different information per sample. Another related line of work has to do with "safe" contextual bandit learning (Kazerouni et al., 2017; Sun et al., 2017). These approaches maintain performance better than a baseline policy at all times, somewhat related to our supervised ground truth setting. However, they do not study the distributional mismatch concerns central to our work.

Finally, there is a substantial literature on active learning from different sources of data (Donmez & Carbonell, 2008; Urner et al., 2012; Yan et al., 2011; Malago et al., 2014; Zhang & Chaudhuri, 2015). The general theme in these papers is to study the setting where the learner has multiple labeling oracles of varying quality, from which it can query labels. Recent work by Yan et al. (2018) studies active learning where additional observational data is given at the beginning of the learning process. The differences between our setting and settings in these works are twofold: first, our setting does not allow algorithms to select examples to query for labels; second, the performance of an algorithm in our setting is measured by its cumulative costs, whereas active learning focuses on optimizing the error of the final classifier.

[1]SUP-ONLY and MAJORITY do not update on CB examples and we plot their final average costs over all CB examples.

2. Notation and Problem Specification

We begin with some notation. For an event A, $I(A) = 1$ if A is true, and 0 otherwise. Denote by $[K]$ the set $\{1, 2, \ldots, K\}$. We use $\mathbb{1}_K$ to denote the all 1's vector in $\mathbb{R}^K$. We use Δ^{K-1} to denote the K dimensional probability simplex.

In this paper, we study the problem of cost-sensitive interactive learning from multiple data sources. Specifically, we consider distributions over cost-sensitive examples (x, c), where $x \in \mathcal{X}$ is a context and c is a cost vector in $[0, 1]^K$; K being the number of actions (or "classes"). There are two distributions D^s (supervised) and D^b (CB), which have identical marginals over the context x, but different conditional distributions over cost vectors given x. We use the notation c^b (resp. c^s) to denote the cost vector c drawn from D^b (resp. D^s) to avoid writing D^b and D^s as subscripts in expectations. The interaction between the learner and the environment is described as follows:

Warm-start: The learner receives S, a dataset of n^s fully supervised examples drawn i.i.d. from D^s.

Interaction: For $t = 1, 2, \ldots, n^b$, the environment draws $(x_t, c_t^b) \sim D^b$ and reveals x_t to the learner, based on which the learner chooses a (possibly random) action $a_t \in [K]$ and observes $c_t^b(a_t)$, but not the cost of any other action.

In this paper, we focus on two learning settings: *CB ground truth setting* and *supervised ground truth setting*. In the CB ground truth setting, the goal of the learner is to optimize the costs drawn from distribution D^b, whereas in the supervised ground truth setting, the goal is to optimize the costs drawn from D^s.

In the content recommendation example (CB ground truth), x_t encodes a user profile and the system predicts which articles (a_t) to display. Here, c^b can be the negative dwell-time of users and c^s is the annotation of editors. The goal of the learner is to optimize the bandit feedback over all displayed articles.

In the translation example (supervised ground truth), x_t encodes the text to be translated and a_t encodes its translation. Here, the goal of the learner is to optimize the costs against the expert translation (c^s) on x_t's, *despite the fact* that the system never sees these costs in its interaction phase. Note that the learner only observes the user feedback costs (c^b) in this interaction phase, which are imperfect proxies for c^s, and the only direct observations of c^s are on the warm-start examples. Nevertheless, we seek to optimize the accuracy of our translations given to the users, and hence regret is still measured over the interaction phase.

To help make decisions, the learner is given a policy class Π that contains elements $\pi : \mathcal{X} \to [K]$. The performance of the algorithm is measured by its *regret* to the retrospective-best policy in Π. We consider two notions of regret over the sequence $\langle x_t \rangle_{t=1}^{n^b}$, depending on whether we consider the CB costs (D^b) or the supervised costs (D^s) to be the ground truth:

$$\text{CB: } \mathbf{R}^b(\langle x_t, a_t \rangle_{t=1}^{n^b}) = \sum_{t=1}^{n^b} \mathbb{E}\big[c^b(a_t) \mid x_t\big] - \min_{\pi\in\Pi} \sum_{t=1}^{n^b} \mathbb{E}\big[c^b(\pi(x_t)) \mid x_t\big], \quad (1)$$

$$\text{supervised: } \mathbf{R}^s(\langle x_t, a_t \rangle_{t=1}^{n^b}) = \sum_{t=1}^{n^b} \mathbb{E}\big[c^s(a_t) \mid x_t\big] - \min_{\pi\in\Pi} \sum_{t=1}^{n^b} \mathbb{E}\big[c^s(\pi(x_t)) \mid x_t\big]. \quad (2)$$

The key difference between the above two regret notions is which cost structure the learner is evaluated against. In the CB ground truth setting (resp. supervised ground truth setting), the regret is defined using the CB cost D^b (resp. supervised cost D^s).

The utility of non ground truth examples are different in the two learning settings. In a CB ground truth setting, relying on the CB examples alone is sufficient to ensure vanishing regret asymptotically. The supervised warm-start primarily helps with a smaller regret in the initial phases of learning. On the other hand, in a supervised ground truth setting, the CB examples can have an asymptotically meaningful effect on the regret: for instance, if $D^s = D^b$, then utilizing CB examples can achieve a vanishing regret, whereas using supervised examples alone cannot.

We can leverage examples from a different source only when the cost structures are at least somewhat related. Therefore, we introduce a measure of similarity of two distributions over cost-sensitive examples.

Definition 1. *Distribution D_2 is (α, Δ)-similar to D_1 wrt policy class Π, if there exists a joint distribution D over $(x, c^+, c^-) \in \mathcal{X} \times [0,1]^K \times [-1,1]^K$ such that $\mathbb{E}_D[c^+(a) + c^-(a)|x] = \mathbb{E}_{D_2}[c(a)|x]$ for all a in $[K]$ and x in $\mathcal{X}$, and there exists a policy $\pi^* \in \operatorname{argmin}_{\pi\in\Pi} \mathbb{E}_{D_1}[c(\pi(x))]$, such that both of the following hold:*

1. *For any policy π in Π, $\mathbb{E}_D[c^+(\pi(x))] - \mathbb{E}_D[c^+(\pi^*(x))] \geq \alpha(\mathbb{E}_{D_1}[c(\pi(x))] - \mathbb{E}_{D_1}[c(\pi^*(x))])$.*
2. *The expected magnitude of c^- is bounded: for any policy π, $|\mathbb{E}_D[c^-(\pi(x))]| \leq \Delta$.*

In the definition above, we use $+/-$ superscripts to qualitatively characterize the utility of c^+, c^-, which give a decomposition of the cost structure generated from D_2. Intuitively, c^+ is the cost structure useful for learning with respect to D_1, as low excess cost with respect to c^+ cost structure implies low excess cost with respect to D_1. In contrast, c^- may or may not have such property. In general, if we have a larger α and smaller Δ, examples from D_2 are more useful for learning with respect to D_1. Prior notions of similarity between distributions for domain adaptation, such as in Ben-David et al. (2010), directly bound the amount of bias in evaluating any $\pi \in \Pi$ under D_1 and D_2. Our definition only requires an upper bound rather than equality for regret under c^+ compared with D_1 and allows an additional scaling factor α, with the idea that we just need to relate the regrets under the two distributions, which we will do shortly. Note that Definition 1 is only used in our analysis; our algorithms do not require knowledge of α and Δ. We now present an example of our similarity notion, followed by a result relating regrets under similar distributions.

Example. Consider a setting where there is a complex cost structure, and collecting warm-start examples by eliciting the full cost vector from an expert is prohibitively expensive. However, an expert can provide an indication of which action has the minimum cost. Formally, let D^b (playing the role of D_1 in Definition 1) be an arbitrary distribution over CB data and let π^* be the best policy in class on D^b. Define D^s (for D_2) by first sampling $(x, c^b) \sim D^b$ and returning (x, c^s) where $c^s(a) = I(a \neq a^*(x))$, where $a^*(x) \in \operatorname{argmin}_a c_b(a)$ is an arbitrary action with the smallest cost according to c_b. Define $c^+(a) = I(a \neq \pi^*(x))$, with $c^- = c^s - c^+$. A direct calculation shows that D^s is $(1, \Delta)$-similar to D^b, with $\Delta = \mathbb{P}(c^b(\pi^*(x)) \geq \min_{a\neq\pi^*(x)} c^b(a))$. [3]

We have the following lemma upper bounding a policy's excess cost on D_1 in terms of its excess cost on D_2:

Lemma 1. *If D_2 is (α, Δ)-similar to D_1 with respect to Π, $\pi^* = \operatorname{argmin}_{\pi\in\Pi} \mathbb{E}_{D_1} c(\pi(x))$, then for any policy π, $\mathbb{E}_{D_2} c(\pi(x)) - \mathbb{E}_{D_2} c(\pi^*(x)) \geq \alpha\left(\mathbb{E}_{D_1} c(\pi(x)) - \mathbb{E}_{D_1} c(\pi^*(x))\right) - 2\Delta$.*

Finally, we define some additional useful notation. Sometimes we will abuse the $\pi(x)$ notation to denote a distribution in Δ^{K-1} induced by (a possibly randomized policy) π over the actions, given instance x. In the t-th round of the interaction process, our algorithms will compute $\hat{c}_t$, an estimate of the unobserved vector c_t^b. We use $\mathbb{E}_S$ to de-

[3] If $c^s(a) = \alpha I(a \neq \pi^*(x))$ for general $\alpha \in [0,1]$, then D^s would be (α, Δ)-similar to D^b.

note sample averages on S and abbreviate $\mathbb{E}_{S_t}$ by $\mathbb{E}_t$ where $S_t = \{(x_\tau, \hat{c}_\tau)\}_{\tau=1}^t$ is the log of the CB examples up to time t.

2.1. Failure of Simple Strategies

The settings we have described so far might appear deceptively simple. After all, it should be straightforward to include some additional supervised examples, which contain strictly more feedback, into a CB algorithm. While this is true to a degree when the two distributions D^s and D^b are well aligned, the potential bias of one source relative to the ground truth makes this task significantly harder.

We now illustrate a simple case of such a failure in the special case of 2-armed bandits (CB with a dummy context), where the CB source is the ground truth. Consider D^s that deterministically yields costs of 0.5 and $0.5 + \frac{\Delta}{2}$ for the two arms, while D^b induces costs 0.5 and $0.5 - \frac{\Delta}{2}$. These two sources are $(1, \Delta)$-similar to each other. Suppose we see $n^s = \Omega(1/\Delta^3)$ examples in warmstart, and use them to initialize the means and confidence intervals on each arm to run the UCB algorithm (Auer et al., 2002a). Using Proposition 1 in Appendix C, we conclude that the optimal arm according to D^b, which is arm 2, is not played even once for the first $O(\exp(1/\Delta))$ rounds, incurring regret $\Omega(\Delta \exp(1/\Delta))$. So for any $\Delta < 0.5$, this scheme incurs a regret strictly larger than that of a UCB algorithm which ignores the warm-start and incurs at most $\tilde{O}(1/\Delta)$ regret.

What we observe here is a failure of competing simultaneously with two baselines. One combines the data from two sources with equal weighting, while the other places a weight of zero on the supervised source. This motivates the solution we present in the next section, where we present algorithms that are competitive with not just these two extremes, but a much larger set of weighted combinations of the two sources. Note that the above example is an illustrative extreme showing that an arbitrary low-regret CB algorithm, when infused with biased warm-start data without additional care, can fail miserably. A simple way to partially ameliorate this problem is through additional uniform exploration, which leads to a faster discovery of the bias, but the added exploration alone is not adequate to use the warm-start data in the most effective manner as we will show in our theory and experiments.

3. Contextual Bandit Ground Truth Setting

In this section, we study the setting where D^b, the distribution over CB examples, is considered the ground truth, as in the content recommendation example from the introduction. Recall that in this setting, one could ignore the supervised warm-start examples entirely and still achieve vanishing regret; the main goal here is to show that using the warm start data can help further reduce the regret, especially in early stages of learning.

3.1. Algorithm

Intuition of our approach. The key challenge in designing an algorithm for the CB ground truth setting is understanding how to effectively combine two data sources which might have unknown differences in their distributions. Before we answer this for the more challenging case where one source is supervised and the other CB, we consider combining two different distributions, but where fully supervised cost vectors are observed in each (and where one is prespecified as the "ground truth" distribution). In this simpler case, Proposition 3 in Appendix H shows that an optimal strategy (up to a constant factor in excess cost) is to exclusively use one source or the other. In particular, if the two sample sizes and the bias Δ are known in advance, then we use samples from the ground truth distribution if and only if its statistical utility exceeds that of the alternative source. However using one source only may be wasteful in practice: if we knew that the two distributions are exactly identical, we would like to find the best policy using all the examples together. A better scheme to combine the data sources, for example, might be to perform loss minimization on a weighted dataset containing examples from both sources.

Based on these insights, we now return to the actual problem setting of warm-starting a CB learner with supervised examples. Our algorithm for this setting is presented in Algorithm 1. The main idea is to minimize the empirical risk on a weighted dataset containing examples from the two sources. Our algorithm picks the mixture weighting by online model selection over a set of weighted combination parameters Λ, where we use the ground truth CB data at each time step to evaluate which $\lambda \in \Lambda$ has the best performance so far. For each $\lambda \in \Lambda$, we estimate a $\pi^\lambda \in \Pi$ as the empirical risk minimizer (ERM) for the λ-mixture between CB and supervised examples. We focus on the simplest ϵ-greedy algorithm for CBs, leaving similar modifications in more advanced CB algorithms for future work.

So long as $\{0, 1\} \subseteq \Lambda$, Algorithm 1 allows for relying on one source alone, while using a larger set of Λ significantly improves its empirical performance (see §5).[4]

In order to present our main technical result about the performance of this algorithm, we need some additional notation. We define $V_t(\lambda)$ that governs the deviation of λ-weighted empirical costs for all policies in Π, and G_t that bounds the

[4] If we approximate the computation of the best policy in Step 6 using an online oracle as in prior works (Agarwal et al., 2014; Langford & Zhang, 2007), then the entire algorithm can be implemented in a streaming fashion since line 7 for selecting the best λ also uses an online estimate a la (Blum et al., 1999) for each λ as opposed to a holdout estimate for the current policy π_t^λ.

Algorithm 1 Adaptively WEighted SOurce MEdley Bandits (AWESOMEBANDITS)

Require: Supervised dataset S from D^s of size n^s, number of interaction rounds n^b, exploration probability ϵ, weighted combination parameters Λ, policy class Π.

for $t = 1, 2, \ldots, n^b$ **do**
  Observe instance x_t from D^b (the same as D^s marginally over x).
  Define $p_t := \frac{1-\epsilon}{t-1} \sum_{\tau=1}^{t-1} \pi_\tau^{\lambda_t}(x_t) + \frac{\epsilon}{K} \mathbb{1}_K$, if $t \geq 2$ and $p_t := \frac{1}{K} \mathbb{1}_K$ for $t = 1$.
  Predict a random label a_t according to the distribution p_t, and receive feedback $c_t^b(a_t)$.
  Define the inverse propensity score (IPS) cost vector $\hat{c}_t(a) := \frac{c_t^b(a_t)}{p_{t,a_t}} I(a = a_t)$, for $a \in [K]$.
  For every $\lambda \in \Lambda$, train π_t^λ as

$$\operatorname*{argmin}_{\pi \in \Pi} \lambda \sum_{\tau=1}^{t-1} \hat{c}_\tau(\pi_\tau^\lambda(x_\tau)) + (1-\lambda) \sum_{(x, c^s) \in S} c^s(\pi(x)). \tag{3}$$

  Set $\lambda_{t+1} \leftarrow \operatorname{argmin}_{\lambda \in \Lambda} \sum_{\tau=1}^{t} \hat{c}_\tau(\pi_\tau^\lambda(x_\tau))$.
end for

excess cost of the ERM solution using weighted combination parameter λ:

$$V_t(\lambda) = 2\sqrt{\left(\lambda^2 \frac{Kt}{\epsilon} + (1-\lambda)^2 n^s\right) \ln \frac{8n^b|\Pi|}{\delta}} + \left(\frac{\lambda K}{\epsilon} + (1-\lambda)\right) \ln \frac{8n^b|\Pi|}{\delta},$$

$$G_t(\lambda, \alpha, \Delta) = \frac{2(1-\lambda)n^s\Delta + 2V_t(\lambda)}{\lambda t + (1-\lambda)n^s\alpha}$$

We establish the following theorem, whose proof is in Appendix E.

Theorem 1. *Suppose D^s is (α, Δ)-similar to D^b. Then for any $\delta < 1/e$, with probability $1 - \delta$, the average CB regret of Algorithm 1 can be bounded as:*

$$\frac{1}{n^b} \mathbf{R}^b(\langle x_t, a_t \rangle_{t=1}^{n^b}) \leq \epsilon + 3\sqrt{\frac{\ln \frac{8n^b|\Pi|}{\delta}}{n^b}} + 32\sqrt{\frac{K \ln \frac{8n^b|\Lambda|}{\delta}}{n^b \epsilon}} + \min_{\lambda \in \Lambda} \frac{\ln(en^b)}{n^b} \sum_{t=1}^{n^b} G_t(\lambda, \alpha, \Delta) \tag{4}$$

The bound (4) consists of many intuitive terms. The first ϵ term comes from uniform exploration; the second term is introduced from bounding the excess expected cost conditioned on x_t's using the unconditional excess expected cost. The next term is the average regret incurred in performing model selection for λ; in our experiments $|\Lambda| = 8$ so that it can be thought of as $O(\sqrt{K/(n^b\epsilon)})$. The final term involving a minimum over λ's is effectively finding the weighted combination which minimizes a bias-variance tradeoff in combining the two sources. Here the bias is controlled by Δ and in place of variance we use the deviation bounds in $V_t(\lambda)$ for high-probability results. Contrasting with learning with CB examples alone, we potentially replace a $\sqrt{\frac{K \ln(|\Pi|/\delta)}{n^b \epsilon}}$ term with the middle term independent of $\ln |\Pi|$ and the average of G_t's which can be much smaller when D^b and D^s are sufficiently similar as we discuss below.

Identical distributions: A very friendly setting has $D^s = D^b$, corresponding to $(1, 0)$-similarity. Since the theorem holds with a minimum over all λ's in the set Λ, we can pick specific values λ_0 of our choice. One choice of λ_0 motivated from prior work (Ben-David et al., 2010) is to pick it such that $\lambda_0/(1-\lambda_0) = \epsilon/K$. This suggests that based on the ratio of the variance of the two sources, one should treat each supervised example as worth K/ϵ CB examples. This setting of $\lambda_0 = \frac{\epsilon}{K+\epsilon}$ yields:

$$G_t(\lambda_0, 1, 0) = O\left(\sqrt{\frac{K \ln \frac{n^b|\Pi|}{\delta}}{\epsilon t + Kn^s}} + \frac{K \ln \frac{n^b|\Pi|}{\delta}}{\epsilon t + Kn^s}\right).$$

Adversarial noise: On the other extreme, we have completely misleading supervised data which is $(1, 1)$-similar to the CB data (we present such an example in the empirical evaluation). In this case, using $\lambda = 1$, we see that $\epsilon + \frac{\ln(en^b)}{n^b} \sum_t G_t(1, 1, 1)$ is the typical regret of the ϵ-greedy algorithm, but there is an additional penalty which is $\widetilde{O}\left(\sqrt{\frac{K \ln |\Lambda|/\delta}{\epsilon n^b}}\right)$ which our algorithm incurs due to the added model selection step. Consequently, it is doubly crucial to avoid extremely small values of ϵ when large and adversarial noise in the warm-start might be anticipated.

Approximate optimality of $\Lambda = \{0, 1\}$: Recall that Proposition 3 shows that when both data sources are fully supervised, it suffices (up to a constant factor in excess cost) to choose $\Lambda = \{0, 1\}$. In our setting where one data source is supervised and the other is CB, Proposition 2 also shows that the individual $G_t(\lambda, \alpha, \Delta)$ terms are approximately minimized at one of these extreme points. But Theorem 1 minimizes an average of these G_t, for which just these two choices might not suffice (since we are likely to prefer $\lambda = 0$ for small t and $\lambda = 1$ for larger t). As we see in the empirical evaluation, using a larger Λ set gives a significant performance boost in many cases.

4. Supervised Ground Truth Setting

In §3, we developed an algorithm and proved regret bounds for combining supervised and CB feedback, in the case where the CB cost is considered the ground truth. In this section, we consider the reverse setting where the supervised source constitutes the ground truth, recalling the motivating example in an automated translation setting from the introduction. Here, we wish to leverage the CB examples for learning the best policy relative to the distribution D^s.

Note that this setting is qualitatively different, since we only have a fixed number n^s of ground-truth examples while the number of CB examples grows over time. If we assign relative weights to individual supervised and CB examples as in Algorithm 1, the CB examples will eventually outweigh the supervised ones for any $\lambda > 0$, which is not desirable when the supervised source is the ground truth. In Algorithm 2, we address this problem by first computing the average cost of every policy on the supervised and CB examples separately, and then choosing a policy that minimizes a weighted combination of them. As a consequence, the relative weight of each CB example diminishes as their number grows, with the overall bias incurred from the CB source staying bounded.

Another difference between Algorithm 2 and Algorithm 1 is that, as opposed to using the CB examples collected online, we use subsets of warm start examples to guide the selection of weighted combination parameter λ. To this end, we introduce an epoch structure in the algorithm. In particular, at each epoch e, λ_e and π_e^{λ}'s are updated exactly once, where a separate validation set is used to pick λ. In addition, we play with uniform randomization around the most recent policy as opposed to a running average of all policies trained so far, an outcome of using a separate validation set (line 12 of Algorithm 2) instead of progressive validation (line 7 of Algorithm 1). Since the exploration policy at the next epoch depends on the previous validation set, we must use a "fresh" validation set at each epoch. Without doing this, we would incur an $\mathcal{O}(\sqrt{\ln|\Pi|/n^s})$ deviation, meaning that we would never outperform learning from supervised data alone. Avoiding this splitting is an interesting question for future work.

For the main result, we need the following notation for the deviation of λ-weighted empirical costs, where $E = \lceil \log n^b \rceil$ is the total number of epochs:

$$W_t(\lambda) = 2\sqrt{\left(\frac{\lambda^2 K}{t\epsilon} + \frac{(1-\lambda)^2(E+1)}{n^s}\right)\ln\frac{8E|\Pi|}{\delta}} + \left(\frac{\lambda K}{t\epsilon} + \frac{(1-\lambda)(E+1)}{n^s}\right)\ln\frac{8E|\Pi|}{\delta}.$$

Theorem 2. *Suppose that D^b is (α, Δ)-similar to D^s. Then for any $\delta < 1/e$, with probability $1-\delta$, the average supervised regret of Algorithm 2 can be bounded as:*

$$\frac{1}{n^b}\mathbf{R}^s(\langle x_t, a_t\rangle_{t=1}^{n^b}) \le \epsilon + 3\sqrt{\frac{\ln\frac{8|\Pi|}{\delta}}{n^b}} + \sqrt{\frac{2(E+1)\ln\frac{8E|\Lambda|}{\delta}}{n^s}} + \min_{\lambda\in\Lambda}\frac{2}{n^b}\sum_{t=1}^{n^b}\frac{2\lambda\Delta + 2W_t(\lambda)}{(1-\lambda)+\lambda\alpha}.$$

Algorithm 2 Combining contextual bandit and supervised data when supervised source is the ground truth

Require: Supervised dataset S from D^s of size n^s, number of interaction rounds n^b, exploration probability ϵ, weighted combination parameters Λ, policy set Π.

1: Let $E = \lceil \log n^b \rceil$ be the number of epochs.
2: Let t_e be the number of CB examples in epoch e: $t_e = \min(2^e, n^b)$ for $e \ge 1$, and $t_0 = 0$.
3: Partition S to $E+1$ equally sized sets $S^{\text{tr}}, S_1^{\text{val}}, \ldots, S_E^{\text{val}}$.
4: **for** $e = 1, 2, \ldots, E$ **do**
5:   **for** $t = t_{e-1}+1, t_{e-1}+2, \ldots, t_e$ **do**
6:     Observe instance x_t from D^b.
7:     Define $p_t := (1-\epsilon)\pi_{e-1}^{\lambda_{e-1}}(x_t) + \frac{\epsilon}{K}\mathbb{1}_K$ for $e \ge 2$, and $p_t := \frac{1}{K}\mathbb{1}_K$ for $e = 1$.
8:     Predict a random label a_t according to the distribution p_t, and receive feedback $c_t^b(a_t)$.
9:     Define the IPS cost vector $\hat{c}_t(a) := \frac{c_t^b(a_t)}{p_{t,a_t}} I(a = a_t)$, for $a \in [K]$.
10:   **end for**
11:   For each $\lambda \in \Lambda$, train π_e^λ as: $\arg\min_{\pi\in\Pi} \lambda\mathbb{E}_{t_e}\hat{c}(\pi(x)) + (1-\lambda)\mathbb{E}_{S^{\text{tr}}} c^s(\pi(x))$.
12:   Set $\lambda_e \leftarrow \arg\min_{\lambda\in\Lambda} \mathbb{E}_{S_e^{\text{val}}} c^s(\pi_e^\lambda(x))$.
13: **end for**

The first term is the cost of exploration, while the second is the gap between the conditional and unconditional expectations over costs in defining the regret. The third term captures the complexity of model selection while the final is the performance upper bound for the best λ in our weighted combination set Λ. As before, this significantly improves upon the $O(\sqrt{\frac{\ln|\Pi|/\delta}{n^s}})$ bound from using supervised examples alone whenever the two sources have sufficient similarity. The proof can be found in Appendix F.

5. Experiments

Experimentally, we focus on the question of learning with the CB costs as the ground truth (§3). Our experiments seek to address the following questions: **a)** How much benefit does the supervised warm-start provide (especially when it is scarce)? **b)** How much benefit does the bandit feedback provide (especially when it is plentiful)? **c)** How robust is our algorithm when there is a realistic mismatch in cost structures? **d)** How robust is our algorithm when the cost structures are highly unaligned (the "safety" question)?

To address these questions, we consider the following set of approaches:

BANDIT-ONLY: a baseline that only uses CB examples.

MAJORITY: always predicts the fixed class which has the smallest cost in expectation according to c^b, without exploration.

SUP-ONLY: a baseline that only uses supervised examples, without exploration.

SIM-BANDIT: a baseline that warm-starts with simulating bandit feedback on the supervised set (using the supervised cost only of the action picked by the CB algorithm during warm start) and then continues on the remaining CB examples.

AWESOMEBANDITS with $\Lambda = \{0, \frac{1}{8}\zeta, \frac{1}{4}\zeta, \frac{1}{2}\zeta, \zeta, \frac{1}{2} + \frac{1}{2}\zeta, \frac{3}{4} + \frac{1}{4}\zeta, 1\}$ (abbrev. **AWESOMEBANDITS with** $|\Lambda| = 8$), where $\zeta = \epsilon/(K+\epsilon)$; this is chosen because ζ is an approximate minimizer of $G_t(\lambda, 1, 0)$, and the $|\Lambda|$ used ensures that $\min_{\lambda\in\Lambda} G_t(\lambda, \alpha, \Delta)$ is close to $\min_{\lambda\in[0,1]} G_t(\lambda, \alpha, \Delta)$ (see Proposition 2). For computational considerations, we use the last policy $\pi_t^{\lambda_t}$ rather than the averaged policy $\frac{1}{t-1}\sum_{\tau=1}^{t-1} \pi_\tau^{\lambda_t}$ in line 3 of Algorithm 1.

MINIMAXBANDITS: the baseline that is equivalent to AWESOMEBANDITS with $\Lambda = \{0, 1\}$. As argued in Proposition 2, choosing λ in this set also approximately minimizes $G_t(\lambda, \alpha, \Delta)$.

All the algorithms (other than SUP-ONLY and MAJORITY, which do not explore) use ϵ-greedy exploration, with most of the results presented using $\epsilon = 0.0125$. We additionally present the results for $\epsilon = 0.1$ and $\epsilon = 0.0625$ in Appendix I. In general, the increased uniform exploration for larger ϵ leads to some performance penalty in the CB algorithms relative to SUP-ONLY, when the bias is small. However, the added exploration gives robustness to large bias as it is readily detected in more adversarial noise settings.

Datasets. We compare these approaches on 524 binary and multiclass classification datasets from Bietti et al. (2018), which in turn are from the `openml.org` platform. For each dataset, we use the original multiclass label in the dataset to generate cost vectors c^b and c^s respectively. That is, given an example $(x, y) \in \mathcal{X} \times [K]$, $c^b(a) = I(a \neq y)$. For each dataset, we vary the number of warm-start examples and CB examples as follows: for a dataset of size n, we vary the number of warm-start examples in $\{0.005n, 0.01n, 0.02n, 0.04n\}$, and the number of CB examples in $\{0.92n, 0.46n, 0.23n, 0.115n\}$. Define the *warm-start ratio* as the ratio of the number of CB examples to the warm-start examples. We group different settings of (dataset, #warm-start examples, #CB examples) by the same warm-start ratio, so that a separate plot is generated for each warm-start ratio in $R = \{2.875, 5.75, 11.5, 23, 46, 92, 184\}$. We filter out the settings where the number of warm-start examples is below 100.

Evaluation Criteria. For each (dataset, #warm-start examples, #CB examples) combination c, we can compute $e_{c,a}$ to be the average cost of algorithm a on the CB examples. Because the range of $e_{c,a}$ can vary significantly over different settings c, we normalize these to yield the *normalized error* of an algorithm on a dataset: $\texttt{err}_{c,a} := \frac{e_{c,a} - e_c^*}{\max_b e_{c,b} - e_c^*}$, where e_c^* is the error achieved by a cost-sensitive one-versus-all learning algorithm trained on all the examples with cost-sensitive label c^b in this dataset, and $\max_b e_{c,b}$ is the maximum error over all algorithms evaluated in this setting. Lower normalized error indicates better performance. We plot the cumulative distribution function (CDF) of the normalized errors for each algorithm. That is, for an algorithm a, at each point x, the y value is the fraction of c's such that $\texttt{err}_{c,a} \leq x$. In general, a high CDF value at a small x indicates that the algorithm is performing well over a large number of (dataset, #warm-start examples, #CB examples) combinations.

In some of the plots when investigating the effect of a particular type or level of noise, we find it useful to aggregate the plots further over all warm-start ratios in creating the CDF. In order to guarantee equal weighting to all noise conditions in our aggregate plots, we combine CDFs for individual ratios through a pointwise average of their CDFs. That is, if $F_{r,a}(x)$ is the fraction of experiments with a given ratio r where the relative error of algorithm a was below x, then the aggregate plot uses $\sum_{r\in R} F_{r,a}(x)/|R|$ as the aggregate CDF for algorithm a.

Comparison with baselines using both sources. We present the CDFs of all algorithms under various noise models in Figure 2, with detailed results for individual noise levels, warm-start ratios and different ϵ values in Appendix I. In Figure 2, we aggregate over warm-start ratios as described earlier. We can see from the figures that AWESOMEBANDITS's CDFs (approximately) dominate those of SIM-BANDIT and MINIMAXBANDITS, which use weightings of 0.5, and the best of $\{0, 1\}$ respectively. These gains highlight the importance of being more careful about selecting a good weighting, despite the earlier intuition from Proposition 3. We see that there is a potentially added benefit of using different λ's in different phases of learning which might even outperform the best setting in hindsight.

Results for aligned cost structures. We conduct experiments in the simple setting where $c^s = c^b$; the CDF plots are in Figure 2a. Here, AWESOMEBANDITS's CDF dominates all other algorithms other than SUP-ONLY. For SUP-ONLY, no exploration is done and the warm-start policy is used greedily, making it a very strong baseline when there is no bias. We observe that AWESOMEBANDITS uses the warm-start much more effectively than both the SIM-BANDIT and MINIMAXBANDITS baselines. Our next experiments consider a uniform at random (UAR) noise setting, where the supervised data is unbiased (with respect to c^b) but has higher variance. In particular, for every example (x, c^b), with probability $1-p$ we set $c^s = c^b$ and with probability p we set

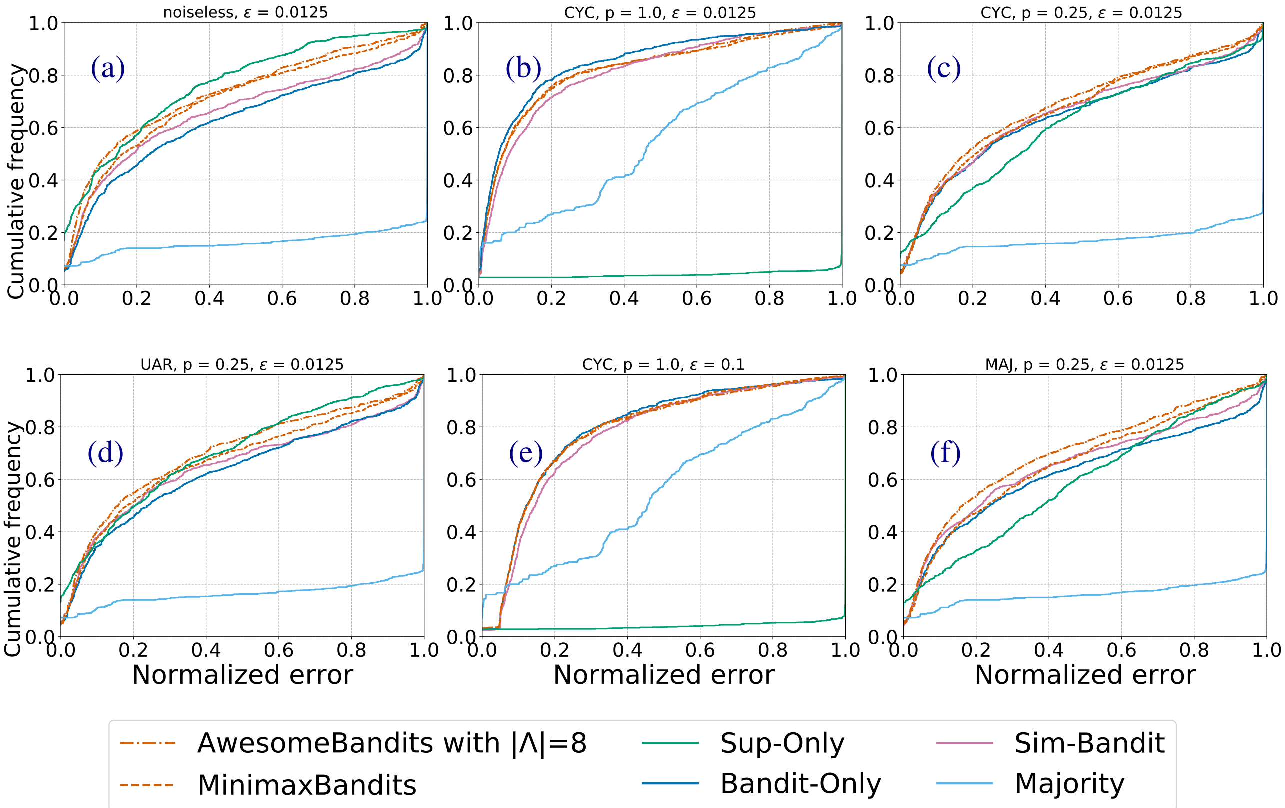


Figure 2: Comparison of all algorithms in the CB ground truth setting using the empirical CDF of the normalized performance scores. Left: unbiased warm-start examples. The settings on the warm start examples are noiseless (top) and UAR with probability 0.5 (down) respectively. Middle: extreme noise rate. The corruption added to the warm-start examples are of type CYC, with probability 1.0. All CB algorithms on the top figure use exploration rate 0.0125, whereas those on the bottom figure use exploration rate 0.1. Right: moderate and potentially helpful noise rates. The corruption added to the warm-start examples are of types CYC (top) and MAJ (down) respectively, with probability 0.25.

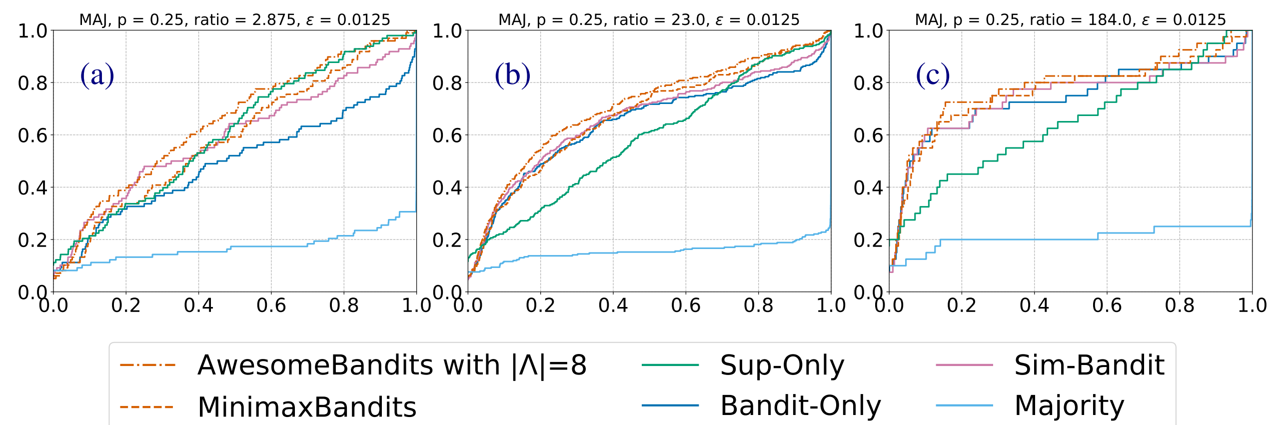


Figure 3: Effect of varying warm-start ratios for MAJ noise with $p = 0.25$. The warm-start ratios vary from 2.875 (left), 23 (middle) to 184 (right). Each CDF aggregates over all conditions with the same warm-start ratio and noise setting in the warm start examples.

c^s as the classification error against a uniform random label. From Claim 1 in the appendix, D^s is $(1-p, 0)$-similar to

D^b. We plot the CDFs of the algorithms in the case where $p = 0.25$ in Figure 2d. The ordering of the CDFs stays essentially the same, with SUP-ONLY less dominant (not surprisingly), and with the gaps between methods reduced with the reduced utility of the warm-start data.

Results with adversarial noise. We next conduct an experiment where c^s and c^b are highly misaligned in order to understand how robust AWESOMEBANDITS is to adversarial conditions. We consider the cycling noise model (CYC), where we set the supervised costs to be "off-by-one" from the CB costs. Specifically, if c^b declares action a to be the zero-cost action, then, with probability p, c^s corrupts the costs so that action $(a + 1) \bmod K$ becomes the zero-cost action. From Claim 2 in the appendix, D^s is $(1, p)$-similar to D^b. The CDF results for this experiment are in Figures 2b, 2e ($p = 1$) for $\epsilon = 0.0125$ and $\epsilon = 0.1$, and Figure 2c ($p = 0.25$) for $\epsilon = 0.0125$ respectively. Again, AWESOMEBANDITS is dominant amongst methods which use both the sources. In the case of $p = 1.0$, BANDIT-ONLY performs the best as the warm start examples are misleading. In this setting, AWESOMEBANDITS performs slightly worse than BANDIT-ONLY for $p = 1$ (Figure 2b). Effectively this gap to BANDIT-ONLY arises due to the additional cost of the model selection step (line 7 in Algorithm 1), as discussed following Theorem 1. This gap is reduced when we increase the ϵ value in ϵ-greedy to 0.1 (Figure 2e). In the case of $p = 0.25$ (Figure 2c), AWESOMEBANDITS outperforms all the methods, showing that it can utilize warm start examples even if they are moderately biased.

Results with majority noise. Finally, we consider the case of a noise model that replaces the ground truth label with the majority label, roughly modeling a "lazy annotator" who occasionally defaults to the most frequent class. For the majority noise model (MAJ), with probability $1 - p$, we set $c^s = c^b$ and with probability p we set c^s to a cost vector that has a zero for the most frequent label in this dataset and one elsewhere. From Claim 2 in the appendix, D^s is $(1, p)$-similar to D^b. The CDFs for this setting are shown in Figure 2f, where we again see AWESOMEBANDITS dominating all the baselines (similar to Figure 2c).

In sum, we observe that AWESOMEBANDITS is the *only* method which is the best or close across all the noise regimes; no other approach is consistently strong. In practical scenarios, where the extent of bias in the warm-start is difficult or costly to ascertain, this robust performance of AWESOMEBANDITS is extremely desirable. If we have some prior information about the noise level, it is prudent to prefer smaller ϵ when we expect a low noise (to compete well with SUP-ONLY), while a larger ϵ is preferred in high noise situations (to quickly detect the extent of bias).

While we only present aggregates over warm-start ratios in the plots here, plots for each combination of noise type, level and warm-start ratio with three different settings of ϵ are shown in Appendix I.

Effect of warm-start ratio. In the plots so far, we have aggregated across the various warm-start ratios for a given noise level. In our next set of experiments (Figure 3), we pick a moderate noise setting and study the ordering of the different methods as the number of warm-start examples increases relative to the CB examples. We see AWESOMEBANDITS outperforming all methods. SUP-ONLY is strong on the left for a small ratio (2.875), while BANDIT-ONLY does well on the other extreme (184), and AWESOMEBANDITS consistently outperform both the baselines combining the two sources.

Overall. Overall, we see that effectively using warm-start examples can certainly improve the performance of CB approaches. AWESOMEBANDITS provides a way to do this in a robust manner, consistently outperforming most baselines. This is best evidenced in Figure 1, which further aggregates performance across the following 10 noise conditions on the warm start examples: noiseless and {UAR, CYC, MAJ} corruptions with probability p in $\{0.25, 0.5, 1.0\}$.

6. Discussion and Future Work

In this paper, we take the first step towards formalizing the question of incorporating multiple data sources in the contextual bandit setting. We see that even in simple settings, obvious techniques do not work robustly, and some care is required to be robust to biases from the non-ground-truth source.

Building on our results, there are several natural avenues for future work. We studied the combination of supervised and bandit information in the context of the ϵ-greedy algorithm. Doing a similar modification to more advanced exploration algorithms (Agrawal & Goyal, 2013; Agarwal et al., 2014, e.g.) is significantly more challenging. This falls into the general category of selecting the best from an ensemble of bandit algorithms (where the ensemble corresponds to different weightings of the supervised and the CB examples), a deceptively difficult task (Agarwal et al., 2017). A possible approach is to employ ideas from their CORRAL algorithm, but the cost of model selection would be linear instead of logarithmic in $|\Lambda|$, and the approach is somewhat data inefficient due to restarts. More ambitiously, it would be desirable for the schedule of supervised and bandit examples to not be fixed in a warm-start fashion but based on active querying by the algorithm, such as by sending uncertain examples to a labeler for full supervision. Studying such interactive settings and considering broader sources of feedback are both interesting avenues for future research.

Acknowledgments.

We thank Alberto Bietti for kindly sharing the scripts for experiments performed by Bietti et al. (2018), and helping getting the experiments running.

References

Agarwal, A., Hsu, D., Kale, S., Langford, J., Li, L., and Schapire, R. Taming the monster: A fast and simple algorithm for contextual bandits. In *International Conference on Machine Learning*, pp. 1638–1646, 2014.

Agarwal, A., Luo, H., Neyshabur, B., and Schapire, R. E. Corralling a band of bandit algorithms. *COLT*, 2017.

Agrawal, S. and Goyal, N. Thompson sampling for contextual bandits with linear payoffs. In *Proceedings of the 30th International Conference on Machine Learning, ICML 2013, Atlanta, GA, USA, 16-21 June 2013*, pp. 127–135, 2013. URL http://jmlr.org/proceedings/papers/v28/agrawal13.html.

Auer, P., Cesa-Bianchi, N., and Fischer, P. Finite-time analysis of the multiarmed bandit problem. *Machine learning*, 47(2-3):235–256, 2002a.

Auer, P., Cesa-Bianchi, N., Freund, Y., and Schapire, R. E. The nonstochastic multiarmed bandit problem. *SIAM J. Comput.*, 32(1):48–77, 2002b. doi: 10.1137/S0097539701398375. URL https://doi.org/10.1137/S0097539701398375.

Ben-David, S., Blitzer, J., Crammer, K., Kulesza, A., Pereira, F., and Vaughan, J. W. A theory of learning from different domains. *Machine Learning*, 79(1-2):151–175, 2010. doi: 10.1007/s10994-009-5152-4. URL https://doi.org/10.1007/s10994-009-5152-4.

Beygelzimer, A., Langford, J., and Zadrozny, B. Weighted one-against-all. In *AAAI*, 2005.

Beygelzimer, A., Langford, J., Li, L., Reyzin, L., and Schapire, R. Contextual bandit algorithms with supervised learning guarantees. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, pp. 19–26, 2011.

Bietti, A., Agarwal, A., and Langford, J. A contextual bandit bake-off. *arXiv preprint arXiv:1802.04064*, 2018.

Blum, A., Kalai, A., and Langford, J. Beating the hold-out: Bounds for k-fold and progressive cross-validation. In *COLT*, 1999.

Chu, W., Li, L., Reyzin, L., and Schapire, R. E. Contextual bandits with linear payoff functions. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics, AISTATS 2011, Fort Lauderdale, USA, April 11-13, 2011*, pp. 208–214, 2011. URL http://www.jmlr.org/proceedings/papers/v15/chu11a/chu11a.pdf.

Crammer, K., Kearns, M., and Wortman, J. Learning from multiple sources. *Journal of Machine Learning Research*, 9:1757–1774, 2008.

Donmez, P. and Carbonell, J. G. Proactive learning: cost-sensitive active learning with multiple imperfect oracles. In *Proceedings of the 17th ACM conference on Information and knowledge management*, pp. 619–628. ACM, 2008.

Duchi, J., Hazan, E., and Singer, Y. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12(Jul):2121–2159, 2011.

Dudik, M., Hsu, D., Kale, S., Karampatziakis, N., Langford, J., Reyzin, L., and Zhang, T. Efficient optimal learning for contextual bandits. In *Proceedings of the 27th Conference on Uncertainty in Artificial Intelligence*. Citeseer, 2011.

Karampatziakis, N. and Langford, J. Online importance weight aware updates. In *Proceedings of the Twenty-Seventh Conference on Uncertainty in Artificial Intelligence*, UAI'11, pp. 392–399, Arlington, Virginia, United States, 2011. AUAI Press. ISBN 978-0-9749039-7-2. URL http://dl.acm.org/citation.cfm?id=3020548.3020594.

Kazerouni, A., Ghavamzadeh, M., Abbasi, Y., and Roy, B. V. Conservative contextual linear bandits. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA*, pp. 3913–3922, 2017. URL http://papers.nips.cc/paper/6980-conservative-contextual-linear-bandits.

Langford, J. and Zhang, T. The epoch-greedy algorithm for contextual multi-armed bandits. In *Proceedings of the 20th International Conference on Neural Information Processing Systems*, pp. 817–824. Curran Associates Inc., 2007.

Malago, L., Cesa-Bianchi, N., and Renders, J. Online active learning with strong and weak annotators. In *NIPS Workshop on Learning from the Wisdom of Crowds*, 2014.

Mansour, Y., Mohri, M., and Rostamizadeh, A. Domain adaptation: Learning bounds and algorithms. *COLT*, 2009.

Nguyen, K., Daumé III, H., and Boyd-Graber, J. Reinforcement learning for bandit neural machine translation with simulated human feedback. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2017. URL http://hal3.name/docs/#daume17simhuman.

Ross, S., Mineiro, P., and Langford, J. Normalized online learning. *UAI*, 2013.

Sokolov, A., Riezler, S., and Urvoy, T. Bandit structured prediction for learning from partial feedback in statistical machine translation. In *MT Summit*, 2015.

Sun, W., Dey, D., and Kapoor, A. Safety-aware algorithms for adversarial contextual bandit. In *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, pp. 3280–3288, 2017. URL http://proceedings.mlr.press/v70/sun17a.html.

Tewari, A. and Murphy, S. A. From ads to interventions: Contextual bandits in mobile health. In *Mobile Health*, pp. 495–517. Springer, 2017.

Urner, R., David, S. B., and Shamir, O. Learning from weak teachers. In *Artificial Intelligence and Statistics*, pp. 1252–1260, 2012.

Yan, S., Chaudhuri, K., and Javidi, T. Active learning with logged data. *ICML*, 2018.

Yan, Y., Rosales, R., Fung, G., and Dy, J. G. Active learning from crowds. In *ICML*, 2011.

Yu, B. Assouad, fano, and le cam. In *Festschrift for Lucien Le Cam*, pp. 423–435. Springer, 1997.

Zhang, C. and Chaudhuri, K. Active learning from weak and strong labelers. In *Advances in Neural Information Processing Systems*, pp. 703–711, 2015.

A. Additional Experimental Details

We run our experiments using Vowpal Wabbit (VW)[5]. In all our algorithms, we consider a scorer function class $\mathcal{F}$ that contains functions f that map (x, a) to estimated cost values. The policy class Π induced by $\mathcal{F}$ is defined as:

$$\Pi := \left\{\pi_f : f \in \mathcal{F}\right\}, \text{ where } \pi_f(x) = \operatorname*{argmin}_{a\in[K]} f(x, a).$$

In general, our algorithms do not learn with respect to Π directly; instead, at each time t, they find some scorer function f_t, and use its induced policy π_{f_t} in Π to perform exploration and exploitation.

In all CB learning algorithms (BANDIT-ONLY and SIM-BANDIT), we use the ϵ-greedy exploration strategy with $\epsilon = 0.0125$ in most of the results for the main text, while the results for two other values (0.1 and 0.0625) in Appendix I. We use the importance weighted regression algorithm (IWR) (Bietti et al., 2018) to compute cost regressors f_t in $\mathcal{F}$. The function class we use consists of linear functions: $f_w(x, a) = \langle w_a, x\rangle$. In the supervised learning algorithm (SUP-ONLY), we use the cost-sensitive one against all algorithm (Beygelzimer et al., 2005) to train a cost regressor f_0 using all the warm start examples, and make no updates in the interaction stage. In the MAJORITY algorithm, we simply predict using the majority class in the dataset and make no updates in the interaction stage. *Note that there is no exploration in* SUP-ONLY *and* MAJORITY. For AWESOMEBANDITS, we also use the same set of ϵ values as BANDIT-ONLY and SIM-BANDIT. At time t and for λ in Λ, we approximate π_t^λ as follows. Instead of optimizing the objective function in Equation (3), we start with finding an approximate optimizer of the following objective function:

$$f_t^\lambda = \operatorname*{argmin}_{f\in\mathcal{F}} \left\{(1-\lambda)\sum_{(x,c)\in S}\sum_{a=1}^{K}(f(x,a) - c(a))^2 + \lambda\sum_{\tau=1}^{t}\frac{1}{p_{\tau,a_\tau}}(f(x_\tau, a_\tau) - c_\tau(a_\tau))^2\right\},$$

and then take $\pi_t^\lambda = \pi_{f_t^\lambda}$. For computational efficiency, we choose not to find the exact empirical cost minimizer f_t^λ at each time. Instead, we use a variant of online gradient descent in VW with adaptive (Duchi et al., 2011), normalized (Ross et al., 2013) and importance-weight aware updates (Karampatziakis & Langford, 2011) on the objective function.

For computational efficiency, in AWESOMEBANDITS, at time $t \geq 2$, instead of mixing the average policy of $\left\{\pi_\tau^\lambda\right\}_{\tau=1}^{t-1}$ with ϵ-uniform exploration, we predict by mixing the most recent policy π_t^λ with ϵ-uniform exploration.

We vary the learning rates of all algorithms from $\{0.1, 0.03, 0.3, 0.01, 1.0, 0.003, 3.0, 0.001, 10.0\}$. For each algorithm and dataset/setting combination, we first compute the average cost in the interaction stage. For different algorithm/dataset/setting combinations, we select different learning rates that minimize the corresponding average cost.

B. Proofs on Similiarity Conditions

Proof of Lemma 1. Suppose (x, c^+, c^-) is the decomposition of (x, c) that satisfies Definition 1. We have that

$$\begin{aligned}\mathbb{E}_{D_2}[c(\pi(x))] - \mathbb{E}_{D_2}[c(\pi^*(x))] &= (\mathbb{E}_D[c^+(\pi(x))] - \mathbb{E}_D[c^+(\pi^*(x))]) + (\mathbb{E}_D[c^-(\pi(x))] - \mathbb{E}_D[c^-(\pi^*(x))]) \\ &\geq \alpha(\mathbb{E}_{D_1}[c(\pi(x))] - \mathbb{E}_{D_1}[c(\pi^*(x))]) - 2\Delta,\end{aligned}$$

where the inequality follows from using the respective items in Definition 1 to lower bound each term (and observing that $\mathbb{E}_D[c^-(\pi(x))] - \mathbb{E}_D[c^-(\pi^*(x))] \leq \Delta + \Delta = 2\Delta$). □

Given a multiclass label y, defined its induced zero-one cost vector $c_y \in [0, 1]^K$ as follows: $c_y(a) = 0$ if $a = y$ and $c_y(a) = 1$ otherwise. From the definition of c_y, it can be seen that for any policy π, $\mathbb{E}c_y(\pi(x)) = \mathbb{P}(\pi(x) \neq y)$.

Claim 1. *Suppose D_0 is a joint distribution over multiclass examples (x, y), and the corrupted label $\tilde{y}$ has the following conditional distribution given (x, y): for all a in $[K]$, $\mathbb{P}(\tilde{y} = a|(x, y)) = (1-p)I(a = y) + \frac{p}{K}$. Define D_1 and D_2 as the joint distribution of (x, c_y) and $(x, c_{\tilde{y}})$, respectively. Then, D_2 is $(1-p, 0)$-similar to D_1.*

Proof. Suppose $\pi^* \in \operatorname{argmin}_{\pi\in\Pi} \mathbb{E}_{D_1} c(\pi(x))$ is an optimal policy with respect to D_1.

[5] http://hunch.net/~vw/

For any policy π, by the definition of $\tilde{y}$ and $c_{\tilde{y}}$, we have that

$$\mathbb{E}c_{\tilde{y}}(\pi(x)) = \mathbb{P}(\pi(x) \neq \tilde{y}) = (1-p)\mathbb{P}(\pi(x) \neq y) + p\frac{K-1}{K}.$$

Therefore, for any policy π, the below identity holds:

$$\mathbb{E}c_{\tilde{y}}(\pi(x)) - \mathbb{E}c_{\tilde{y}}(\pi^*(x)) = (1-p)(\mathbb{E}c_y(\pi(x)) - \mathbb{E}c_y(\pi^*(x))).$$

Therefore, taking $c^+ = c_{\tilde{y}}$, $c^- = 0$, $\alpha = 1-p$ and $\Delta = 0$, it can be easily checked that the conditions of Definition 1 are satisfied. □

Claim 2. *Suppose D_0 is a joint distribution over multiclass examples (x, y), and the corrupted label $\tilde{y}$'s conditional distribution given (x, y) has the following property: $\mathbb{P}(\tilde{y} = y|(x, y)) \geq 1-p$. Define D_1 and D_2 as the joint distribution of (x, c_y) and $(x, c_{\tilde{y}})$, respectively. Then, D_2 is $(1, p)$-similar to D_1.*

Proof. Suppose $\pi^* \in \operatorname{argmin}_{\pi\in\Pi} \mathbb{E}_{D_1} c(\pi(x))$ is an optimal policy with respect to D_1.

For every x, define deterministically that $c^+ = \mathbb{E}[c_y|x]$, and $c^- = \mathbb{E}[c_{\tilde{y}}|x] - \mathbb{E}[c_y|x]$. We have that $\mathbb{E}[c^+ + c^-|x] = \mathbb{E}[c_{\tilde{y}}|x]$ by the definitions of c^+ and c^-. In addition, by the construction of c^+, we immediately have that

$$\mathbb{E}c^+(\pi(x)) - \mathbb{E}c^+(\pi^*(x)) = \mathbb{E}c_y(\pi(x)) - \mathbb{E}c_y(\pi^*(x)).$$

What remains is to bound $\mathbb{E}c^-(\pi(x))$. By the definitions of c_y and $c_{\tilde{y}}$, we have that

$$|\mathbb{E}c^-(\pi(x))| = |\mathbb{P}(\pi(x) \neq \tilde{y}) - \mathbb{P}(\pi(x) \neq y)| \leq \mathbb{P}(y \neq \tilde{y}).$$

By the assumption on the conditional distribution of $\tilde{y}$ given (x, y), we have that the right hand side is at most p. The claim follows. □

C. Proof Showing the Failure of Equal Data Weighting

In this section we formalize the example presented in §2.1. To recall, this is a 2-armed bandit setting (i.e. contextual bandits with a dummy context), where policy class $\Pi := \{\pi_1, \pi_2\}$, where π_i maps any context to action i, $i = 1, 2$. In addition, D^s (resp. D^b) is the Dirac measure on (x_0, c_0^s) (resp. (x_0, c_0^b)), and the respective cost vectors are:

$$c_0^s = (0.5, 0.5 + \frac{\Delta}{2}), \quad \text{and} \quad c_0^b = (0.5, 0.5 - \frac{\Delta}{2}).$$

We consider the following algorithm, which directly extends the UCB1 algorithm (Auer et al., 2002a) by additionally using the warm start examples to estimate the mean costs of the two actions. Note that as it is minimizing its cumulative cost, the algorithm computes lower confidence bounds of the costs and selects the minimum, which is equivalent to computing upper confidence bounds of the rewards and selecting the maximum.

Algorithm 3 A variant of the UCB1 algorithm that accounts for warm start examples

Require: Supervised examples $S = \{(x_0, c^s)\}$ of size n^s, number of interaction rounds n^b.

for $t = 1, 2, \ldots, n^b$ **do**

  For $i = 1, 2$, define $n_{i,t-1} = \sum_{s=1}^{t-1} I(a_t = i)$.

  For $i = 1, 2$, compute empirical mean cost of action i: $\hat{\mu}_{i,t} = \frac{\sum_{(x_0,c^s)\in S} c^s(i) + \sum_{s=1}^{t-1} I(a_t=i)c_s^b(i)}{n^s + n_{i,t-1}}$.

  For $i = 1, 2$, compute $\text{lcb}_{i,t} = \hat{\mu}_{i,t} - 2\sqrt{\frac{\ln t}{n^s + n_{i,t-1}}}$.

  Take action $a_t = \operatorname{argmin}_{i\in\{1,2\}} \text{lcb}_{i,t}$.

  Observe $c_t^b(a_t)$.

end for

We have the following proposition on a lower bound of the regret of Algorithm 3, under the above settings of D^s and D^b.

Proposition 1. *Suppose D^s and D^b are described as above. Additionally, Algorithm 3 is run with input n^s warm start examples drawn from D^s and number of interaction rounds $n^b \geq \exp(\Delta^2 n^s/16)$. Then, Algorithm 3 incurs a regret of* $\Omega(\Delta \exp(\Delta^2 n_s/16))$.

Proof. Suppose after $t-1$ rounds of the interaction phase, Algorithm 3 has taken action i t_i times for $i = 1, 2$. As the cost vectors are deterministic, we can calculate the lower confidence bound estimates for the two actions in closed form:

$$\mathrm{lcb}_{t,1} = 0.5 - 2\sqrt{\frac{\ln(t_1 + t_2 + 1)}{n^s + t_1}} \quad \text{and} \quad \mathrm{lcb}_{t,2} = 0.5 + \frac{\Delta}{2}\frac{n^s - t_2}{n^s + t_2} - 2\sqrt{\frac{\ln(t_1 + t_2 + 1)}{n^s + t_2}}$$

Now, let us consider the first time we play action 2 in the interaction phase. At this point, t_2 is still 0 and t_1 should satisfy

$$\mathrm{lcb}_{t,2} \leq \mathrm{lcb}_{t,1}, \quad \text{that is} \quad \frac{\Delta}{2} - 2\sqrt{\frac{\ln(t_1+1)}{n^s}} \leq -2\sqrt{\frac{\ln(t_1+1)}{n^s + t_1}}.$$

The above condition implies that

$$2\sqrt{\frac{\ln(t_1+1)}{n^s}} \geq \frac{\Delta}{2}, \quad \text{equivalently} \quad t_1 \geq \exp(\Delta^2 n^s/16) - 1.$$

Denote by $T_0 := \exp(\Delta^2 n^s/16) - 1$. Therefore, after $n^b \geq \exp(\Delta^2 n^s/16)$ rounds of interaction, the regret of Algorithm 3 can be lower bounded by:

$$\sum_{t=1}^{n^b} c_0^b(a_t) - c_0^b(2) \geq \sum_{t=1}^{T_0 - 1} c_0^b(1) - c_0^b(2) = \frac{\Delta}{2} \cdot (T_0 - 1) = \Omega(\Delta \exp(\Delta^2 n^s/16)). \qquad \square$$

In Algorithm 3, we used only $\ln(t+1)$ in the numerator, when an alternative might be to use $\ln(t + n^s + 1)$. However, it is easily checked that after this simple modification, a similar exponential regret lower bound of Algorithm 3 can be proved (with the definition of T_0 changed to $T_0 := \exp(\Delta^2 n^s/16) - n^s - 1$).

D. Concentration Inequalities

We use a version of Freedman's inequality from (Beygelzimer et al., 2011).

Lemma 2 (Freedman's inequality)**.** *Let $X_1, \ldots, X_n$ be a martingale difference sequence adapted to filtration $\{\mathcal{B}_i\}_{i=0}^n$, and $|X_i| \leq M$ almost surely for all i. Let $V = \sum_{i=1}^n \mathbb{E}[X_i^2 | \mathcal{B}_{i-1}]$ be the cumulative conditional variance. Then, with probability $1 - \delta$,*

$$\left|\sum_{i=1}^n X_i\right| \leq 2\sqrt{V \ln \frac{2}{\delta}} + M \ln \frac{2}{\delta}.$$

E. Proof of Theorem 1

We begin with some additional notation used in the analysis. Throughout this section, we let $\pi^* = \operatorname{argmin}_{\pi\in\Pi} \mathbb{E}c^b(\pi(x))$, the optimal policy in Π with respect to D^b.

Recall that for policy π and λ in $[0, 1]$, we define $\mathbb{E}_t[\hat{c}(\pi(x))] := \frac{1}{t}\sum_{s=1}^t \hat{c}_s(\pi(x))$ and $\mathbb{E}_S[c^s(\pi(x))] := \frac{1}{n^s}\sum_{(x,c)\in S} c(\pi(x))$. We additionally define the λ-weighted empirical cost of π as

$$\hat{L}_{\lambda,t}(\pi) = \frac{\lambda t \mathbb{E}_t[\hat{c}(\pi(x))] + (1-\lambda) n^s \mathbb{E}_S[c^s(\pi(x))]}{\lambda t + (1-\lambda) n^s},$$

and its expectation

$$L_{\lambda,t}(\pi) = \frac{\lambda t \mathbb{E}[c^b(\pi(x))] + (1-\lambda) n^s \mathbb{E}[c^s(\pi(x))]}{\lambda t + (1-\lambda) n^s}.$$

Observe that $\hat{L}_{1,t}(\pi) = \mathbb{E}_t[\hat{c}(\pi(x))]$ is π's empirical cost on the first t CB examples, $L_{1,t}(\pi) = \mathbb{E}c^b(\pi(x))$, $\hat{L}_{0,t}(\pi) = \mathbb{E}_S[c^s(\pi(x))]$ is π's empirical cost on the n^s supervised examples, and $L_{0,t}(\pi) = \mathbb{E}c^s(\pi(x))$. Denote by $\hat{\Gamma}_{\lambda,t}(\pi) = (\lambda t + (1-\lambda)n^s)\hat{L}_{\lambda,t}(\pi)$ the unnormalized λ-weighted empirical cost of π, and $\Gamma_{\lambda,t}(\pi) = (\lambda t + (1-\lambda)n^s)L_{\lambda,t}(\pi)$ its expectation.

Denote by $(x_{n^b+1}, c^s_{n^b+1}), \ldots, (x_{n^b+n^s}, c^s_{n^b+n^s})$ an enumeration of the elements in S. Define filtration $\{\mathcal{B}_t\}_{t=0}^{n^b+n^s}$ as follows: $\mathcal{B}_0$ is the trivial σ-algebra, and

$$\mathcal{B}_t = \begin{cases} \sigma((x_{n^b+1}, c^s_{n^b+1}), \ldots, (x_{n^b+1}, c^s_{n^b+t})), & t \in \{1, \ldots, n^s\}, \\ \sigma(S, (x_1, \hat{c}_1), \ldots, (x_{t-n^s}, \hat{c}_{t-n^s})), & t \in \{n^s+1, \ldots, n^s+n^b\}. \end{cases}$$

For reader's convenience, we also recall our earlier notation:

$$V_t(\lambda) = 2\sqrt{\left(\lambda^2 \tfrac{Kt}{\epsilon} + (1-\lambda)^2 n^s\right) \ln \tfrac{8n^b|\Pi|}{\delta}} + \left(\tfrac{\lambda K}{\epsilon} + (1-\lambda)\right) \ln \tfrac{8n^b|\Pi|}{\delta},$$
$$G_t(\lambda, \alpha, \Delta) = \tfrac{2(1-\lambda)n^s\Delta + 2V_t(\lambda)}{\lambda t + (1-\lambda)n^s\alpha}.$$

In addition, denote by

$$\bar{G}_t(\lambda, \alpha, \Delta) := \min(1, G_t(\lambda, \alpha, \Delta)).$$

Proof of Theorem 1. Define event I as: for all π in Π,

$$\left|\frac{1}{n^b}\sum_{t=1}^{n^b}(\mathbb{E}[c^b_t(\pi(x_t))|x_t] - \mathbb{E}c^b(\pi(x)))\right| \le \sqrt{\frac{\ln\frac{8|\Pi|}{\delta}}{2n^b}}.$$

By Hoeffding's inequality and union bound, I happens with probability $1-\frac{\delta}{4}$. Specifically, on event I, for every policy π in Π, as π^* is the policy in Π that minimizes $\mathbb{E}c^b(\pi(x))$, we have

$$\mathbb{E}c^b(\pi^*(x)) - \frac{1}{n^b}\sum_{t=1}^{n^b}\mathbb{E}[c^b(\pi(x_t))|x_t] \le \mathbb{E}c^b(\pi(x)) - \frac{1}{n^b}\sum_{t=1}^{n^b}\mathbb{E}[c^b(\pi(x_t))|x_t] \le \sqrt{\frac{\ln\frac{8|\Pi|}{\delta}}{2n^b}}.$$

Therefore,

$$\mathbb{E}c^b(\pi^*(x)) - \min_{\pi\in\Pi}\frac{1}{n^b}\sum_{t=1}^{n^b}\mathbb{E}[c^b(\pi(x_t))|x_t] \le \sqrt{\frac{\ln\frac{8|\Pi|}{\delta}}{2n^b}}. \tag{5}$$

Recall that randomized policy $\pi_t : \mathcal{X} \to \Delta^{K-1}$ is defined as $\pi_t(x) = \frac{1-\epsilon}{t-1}\sum_{\tau=1}^{t-1}\pi_\tau^{\lambda_t}(x) + \frac{\epsilon}{K}\mathbb{1}_K$ for $t \ge 2$, and $\pi_1(x) = \frac{1}{K}\mathbb{1}_K$ for all x. With a slight abuse of notation, denote by $\mathbb{E}c^b(\pi_t(x)) := \mathbb{E}_{(x,c^b),a\sim\pi_t(x)}c^b(a)$. Observe that $\mathbb{E}[\mathbb{E}[c^b(a_t)|x_t]|\mathcal{B}_{n^s+t-1}] = \mathbb{E}c^b(\pi_t(x))$. Define event J as:

$$\left|\frac{1}{n^b}\sum_{t=1}^{n^b}\mathbb{E}[c^b(a_t)|x_t] - \frac{1}{n^b}\sum_{t=1}^{n^b}\mathbb{E}c^b(\pi_t(x))\right| \le \sqrt{\frac{\ln\frac{8}{\delta}}{2n^b}}. \tag{6}$$

By Azuma's inequality, J happens with probability $1-\frac{\delta}{4}$.

Denote by E the event that the events E_t, F_t defined in Lemmas 3 and 5 (both given below) and I, J hold simultaneously for all t. By a union bound over all E_t, F_t's and I, J, event E happens with probability $1-\delta$. We henceforth condition on E happening.

Consider t in $\{2, \ldots, n^b\}$. We now give an upper bound on the expected excess cost of using randomized prediction π_t. Observe that by the definition of π_t,

$$\mathbb{E}c^b(\pi_t(x)) = \frac{1-\epsilon}{t-1}\sum_{\tau=1}^{t-1}\mathbb{E}c^b(\pi_\tau^{\lambda_t}(x)) + \epsilon\frac{1}{K}\sum_{a=1}^{K}\mathbb{E}c^b(a) \le \frac{1}{t-1}\sum_{\tau=1}^{t-1}\mathbb{E}c^b(\pi_\tau^{\lambda_t}(x)) + \epsilon, \tag{7}$$

it suffices to upper bound $\frac{1}{t-1}\sum_{\tau=1}^{t-1}\mathbb{E}c^b(\pi_\tau^{\lambda_t}(x))$.

By Lemma 5 below, we have that

$$\frac{1}{t-1}\sum_{\tau=1}^{t-1}\mathbb{E}c^b(\pi_\tau^{\lambda_t}(x)) - \min_{\lambda\in\Lambda}\frac{1}{t-1}\sum_{\tau=1}^{t-1}\mathbb{E}c^b(\pi_\tau^{\lambda}(x)) \le 16\sqrt{\frac{K\ln\frac{8n^b|\Lambda|}{\delta}}{(t-1)\epsilon}}. \tag{8}$$

In the above inequality, we can bound the individual summands of the second term by Lemma 3:

$$\mathbb{E}c^b(\pi_\tau^{\lambda}(x)) - \mathbb{E}c^b(\pi^*(x)) \le \bar{G}_{\tau-1}(\lambda,\alpha,\Delta).$$

Therefore, rewriting Equation (8), we get that

$$\frac{1}{t-1}\sum_{\tau=1}^{t-1}\mathbb{E}c^b(\pi_\tau^{\lambda_t}(x)) - \min_{\lambda\in\Lambda}\left[\frac{1}{t-1}\sum_{\tau=1}^{t-1}(\mathbb{E}c^b(\pi^*(x)) + \bar{G}_{\tau-1}(\lambda,\alpha,\Delta))\right] \le 16\sqrt{\frac{K\ln\frac{8n^b|\Lambda|}{\delta}}{(t-1)\epsilon}}.$$

Combining the above with Equation (7), we get that

$$\mathbb{E}c^b(\pi_t(x)) - \mathbb{E}c^b(\pi^*(x)) \le \epsilon + \min_{\lambda\in\Lambda}\left[\frac{1}{t-1}\sum_{\tau=1}^{t-1}\bar{G}_{\tau-1}(\lambda,\alpha,\Delta)\right] + 16\sqrt{\frac{K\ln\frac{8n^b|\Lambda|}{\delta}}{(t-1)\epsilon}}.$$

Summing the above inequality over t in $\{2,\ldots,n^b\}$ and using that $\mathbb{E}c^b(\pi_1(x)) - \mathbb{E}c^b(\pi^*(x)) \le 1$, we get,

$$\sum_{t=1}^{n^b}[\mathbb{E}c^b(\pi_t(x)) - \mathbb{E}c^b(\pi^*(x))] \le n^b\epsilon + \sum_{t=2}^{n^b}\min_{\lambda\in\Lambda}\left[\frac{1}{t-1}\sum_{\tau=1}^{t-1}\bar{G}_{\tau-1}(\lambda,\alpha,\Delta)\right] + 16\sum_{t=2}^{n^b}\sqrt{\frac{K\ln\frac{8n^b|\Lambda|}{\delta}}{(t-1)\epsilon}} + 1.$$

Observe that for any function $\{N_t(\cdot)\}_{t=1}^{n^b}$, $\sum_{t=1}^{n^b}\min_{\lambda\in\Lambda}N_t(\lambda) \le \min_{\lambda\in\Lambda}[\sum_{t=1}^{n^b}N_t(\lambda)]$. We further collect the coefficients on the $\bar{G}_\tau(\lambda,\alpha,\Delta)$ terms and use the upper bound $\sum_{t=\tau+1}^{n^b}\frac{1}{t-1} \le \ln(en^b)$ for all $\tau \ge 1$, getting

$$\sum_{t=1}^{n^b}[\mathbb{E}c^b(\pi_t(x)) - \mathbb{E}c^b(\pi^*(x))] \le n^b\epsilon + \ln(en^b)\cdot\min_{\lambda\in\Lambda}\sum_{\tau=0}^{n^b-2}\bar{G}_\tau(\lambda,\alpha,\Delta) + 16\sum_{t=2}^{n^b}\sqrt{\frac{K\ln\frac{8n^b|\Lambda|}{\delta}}{(t-1)\epsilon}} + 1. \tag{9}$$

Therefore, we get:

$$\begin{aligned}
& \mathbf{R}^b(\langle x_t, a_t\rangle_{t=1}^{n^b}) \\
\le\ & \frac{1}{n^b}\sum_{t=1}^{n^b}[\mathbb{E}c^b(\pi_t(x)) - \mathbb{E}c^b(\pi^*(x))] + \sqrt{\frac{2\ln\frac{8|\Pi|}{\delta}}{n^b}} \\
\le\ & \epsilon + \frac{\ln(en^b)}{n^b}\min_{\lambda\in\Lambda}\sum_{t=0}^{n^b-2}\bar{G}_t(\lambda,\alpha,\Delta) + \frac{16}{n^b}\sum_{t=2}^{n^b}\sqrt{\frac{K\ln\frac{8n^b|\Lambda|}{\delta}}{(t-1)\epsilon}} + \sqrt{\frac{2\ln\frac{8|\Pi|}{\delta}}{n^b}} + \frac{1}{n^b} \\
\le\ & \epsilon + \frac{\ln(en^b)}{n^b}\min_{\lambda\in\Lambda}\sum_{t=1}^{n^b}\bar{G}_t(\lambda,\alpha,\Delta) + \frac{16}{n^b}\sum_{t=2}^{n^b}\sqrt{\frac{K\ln\frac{8n^b|\Lambda|}{\delta}}{(t-1)\epsilon}} + \sqrt{\frac{2\ln\frac{8|\Pi|}{\delta}}{n^b}} + \frac{\ln(e^2n^b)}{n^b} \\
\le\ & \epsilon + 3\sqrt{\frac{\ln\frac{8n^b|\Pi|}{\delta}}{n^b}} + 32\sqrt{\frac{K\ln\frac{8n^b|\Lambda|}{\delta}}{n^b\epsilon}} + \frac{\ln(en^b)}{n^b}\min_{\lambda\in\Lambda}\sum_{t=1}^{n^b}\bar{G}_t(\lambda,\alpha,\Delta).
\end{aligned}$$

where the first inequality is from Equations (5) and (6) and dividing both sides by n^b; the second inequality is from Equation (9); the third inequality is from that $\bar{G}_0(\lambda,\alpha,\Delta) \le 1$; the fourth inequality is from algebra, and our assumption that $\delta < 1/e$. The theorem follows. □

The following lemma upper bounds the excess cost of $\pi_t^\lambda = \operatorname{argmin}_{\pi\in\Pi} \hat{L}_{\lambda,t-1}(\pi)$.

Lemma 3. *For every $t \in \{1,\ldots,n^b\}$, there exists an event E_t with probability $1-\frac{\delta}{4n^b}$, such that the following holds for all λ in Λ:*

$$\mathbb{E}c^b(\pi_t^\lambda(x)) - \mathbb{E}c^b(\pi^*(x)) \le \bar{G}_{t-1}(\lambda,\alpha,\Delta).$$

Proof. Define event E_t as: for all π in Π,

$$\begin{aligned} &\Big|[\lambda(t-1)\mathbb{E}c^b(\pi(x)) + (1-\lambda)n^s\mathbb{E}c^s(\pi(x))] - [\lambda(t-1)\mathbb{E}_{t-1}\hat{c}(\pi(x)) + (1-\lambda)n^s\mathbb{E}_S c^s(\pi(x))]\Big| \\ \le\quad & 2\sqrt{(\lambda^2\frac{K(t-1)}{\epsilon} + (1-\lambda)^2 n^s)\ln\frac{8n^b|\Pi|}{\delta}} + (\frac{\lambda K}{\epsilon} + (1-\lambda))\ln\frac{8n^b|\Pi|}{\delta} \end{aligned} \tag{10}$$

In other words,

$$\Big|\Gamma_{\lambda,t-1}(\pi) - \hat{\Gamma}_{\lambda,t-1}(\pi)\Big| \le V_{t-1}(\lambda). \tag{11}$$

For a fixed π, applying Lemma 2 with $X_i = (1-\lambda)(c^s_{n^b+i}(\pi(x_{n^b+i}) - \mathbb{E}c^s(\pi(x)))$ for i in $\{1,\ldots,n^s\}$, $X_i = \lambda(\hat{c}_{i-n^s}(\pi(x_{i-n^s})) - \mathbb{E}c^b(\pi(x)))$ for i in $\{n^s+1,\ldots,n^s+t\}$, and $M = (1-\lambda) + \frac{\lambda K}{\epsilon}$, and noting that $|X_i| \le M$ almost surely for all i, $\mathbb{E}[X_i^2|\mathcal{B}_{i-1}] \le (1-\lambda)^2$ for i in $\{1,\ldots,n^s\}$, and $\mathbb{E}[X_i^2|\mathcal{B}_{i-1}] \le \lambda^2\mathbb{E}[\frac{1}{p_{i-n^s,\pi(x_{i-n^s})}}|\mathcal{B}_{i-1}] \le \frac{\lambda^2 K}{\epsilon}$ for i in $\{n^s+1,\ldots,n^s+t-1\}$, we get that Equation (10) holds for π with probability $1-\frac{\delta}{4n^b|\Pi|}$. Therefore, by an union bound over all π in Π, E_t happens with probability $1-\frac{\delta}{4n^b}$. We henceforth condition on E_t happening.

By the optimality of π_t^λ,

$$\hat{L}_{\lambda,t-1}(\pi_t^\lambda) \le \hat{L}_{\lambda,t-1}(\pi^*).$$

Equivalently,

$$\hat{\Gamma}_{\lambda,t-1}(\pi_t^\lambda) \le \hat{\Gamma}_{\lambda,t-1}(\pi^*).$$

Combining with Equation (11) applied to π_t^λ and π^*, we get that

$$\Gamma_{\lambda,t-1}(\pi_t^\lambda) - \Gamma_{\lambda,t-1}(\pi^*) \le 2V_{t-1}(\lambda).$$

Using the (α,Δ)-similarity of D^s to D^b, and Lemma 4 below, we get that

$$(\mathbb{E}c^b(\pi_t^\lambda(x)) - \mathbb{E}c^b(\pi^*(x)))(\lambda(t-1) + (1-\lambda)n^s\alpha) \le 2V_{t-1}(\lambda,\alpha) + 2(1-\lambda)n^s\Delta.$$

Therefore, by the definition of $G_t(\lambda,\alpha,\Delta)$, we have that $\mathbb{E}c^b(\pi_t^\lambda(x)) - \mathbb{E}c^b(\pi^*(x))) \le G_{t-1}(\lambda,\alpha,\Delta)$. Combining the above with the fact that $\mathbb{E}c^b(\pi_t^\lambda(x)) - \mathbb{E}c^b(\pi^*(x)) \le 1$, the lemma follows. □

Lemma 4. *If D^s is (α,Δ)-similar to D^b, then for any policy π,*

$$(\mathbb{E}c^b(\pi(x)) - \mathbb{E}c^b(\pi^*(x)))(\lambda t + (1-\lambda)n^s\alpha) \le (\Gamma_{\lambda,t}(\pi) - \Gamma_{\lambda,t}(\pi^*)) + 2(1-\lambda)n^s\Delta.$$

Proof. Using the definition of $\Gamma_{\lambda,t}$, we have that

$$\Gamma_{\lambda,t}(\pi) - \Gamma_{\lambda,t}(\pi^*) = \lambda t(\mathbb{E}c^b(\pi(x)) - \mathbb{E}c^b(\pi^*(x))) + (1-\lambda)n^s(\mathbb{E}c^s(\pi(x)) - \mathbb{E}c^s(\pi^*(x))).$$

Applying Lemma 1, the right hand side is at least

$$\lambda t(\mathbb{E}c^b(\pi(x)) - \mathbb{E}c^b(\pi^*(x))) + (1-\lambda)n^s(\alpha(\mathbb{E}c^b(\pi(x)) - \mathbb{E}c^b(\pi^*(x))) - 2\Delta).$$

The lemma follows immediately by algebra. □

We also bound the cost overhead for selecting λ_t from Λ compared to using the best λ in hindsight. Define the progressive validation error using $\{\pi_\tau^\lambda\}_{\tau\le t}$ as: $\hat{C}_{\lambda,t} = \sum_{\tau=1}^t \hat{c}_\tau(\pi_\tau^\lambda(x_\tau))$, and its expectation as: $C_{\lambda,t} = \sum_{\tau=1}^t \mathbb{E}c^b(\pi_\tau^\lambda(x))$.

Lemma 5. *For every $t \in \{2, \dots, n^b\}$, there exists an event F_t with probability $1 - \frac{\delta}{4n^b}$, such that λ_t has the following property:*

$$\sum_{\tau=1}^{t-1} \mathbb{E} c^b(\pi_\tau^{\lambda_t}(x)) - \min_{\lambda \in \Lambda} \sum_{\tau=1}^{t-1} \mathbb{E} c^b(\pi_\tau^{\lambda}(x)) \le 16\sqrt{\frac{K(t-1)}{\epsilon} \ln \frac{8n^b|\Lambda|}{\delta}}.$$

Proof. Define event F_t as: for all λ in Λ,

$$\left| C_{\lambda,t-1} - \hat{C}_{\lambda,t-1} \right| \le 2\sqrt{\frac{K(t-1)}{\epsilon} \ln \frac{8n^b|\Lambda|}{\delta}} + \frac{K}{\epsilon} \ln \frac{8n^b|\Lambda|}{\delta}.$$

In other words,

$$\left| \sum_{\tau=1}^{t-1} \mathbb{E} c^b(\pi_\tau^{\lambda}(x)) - \sum_{\tau=1}^{t-1} \hat{c}_\tau(\pi_\tau^{\lambda}(x_\tau)) \right| \le 2\sqrt{\frac{K(t-1)}{\epsilon} \ln \frac{8n^b|\Lambda|}{\delta}} + \frac{K}{\epsilon} \ln \frac{8n^b|\Lambda|}{\delta}. \tag{12}$$

For a fixed λ, by Lemma 2, taking $X_i = 0$ for i in $[n^s]$, $X_i = \hat{c}_{i-n^s}(\pi_{i-n^s}^{\lambda}(x_{i-n^s})) - \mathbb{E} c^b(\pi_{i-n^s}^{\lambda}(x))$ for i in $\{n^s+1, \dots, n^s+t-1\}$, $M = \frac{K}{\epsilon}$, and noting that $|X_i| \le M$ almost surely for all i, $\mathbb{E}[X_i^2|\mathcal{B}_{i-1}] = 0$ for i in $[n^s]$, and $\mathbb{E}[X_i^2|\mathcal{B}_{i-1}] \le \mathbb{E}[\frac{1}{p_{i-n^s,\pi(x_{i-n^s})}}|\mathcal{B}_{i-1}] \le \frac{K}{\epsilon}$ for i in $\{n^s+1, \dots, n^s+t-1\}$, we get that Equation (12) holds with probability $1 - \frac{\delta}{4n^b|\Lambda|}$. By an union bound over all λ in Λ, the probability of F_t is at least $1 - \frac{\delta}{4n^b}$. We henceforth condition on F_t happening.

By the optimality of λ_t, we know that for all λ in Λ,

$$\hat{C}_{\lambda_t,t-1} \le \hat{C}_{\lambda,t-1}.$$

Combining the above inequality with Equation (12) applied on λ_t and λ, we get that

$$C_{\lambda_t,t-1} - C_{\lambda,t-1} \le 4\sqrt{\frac{K(t-1)}{\epsilon} \ln \frac{8n^b|\Lambda|}{\delta}} + 2\frac{K}{\epsilon} \ln \frac{8n^b|\Lambda|}{\delta}.$$

In addition, observe that $C_{\lambda_t,t-1} - C_{\lambda,t-1} \le t-1$ as $c^b \in [0,1]^K$ with probability 1. Combining the above facts with Lemma 6 below, we have that

$$C_{\lambda_t,t-1} - C_{\lambda,t-1} \le \min\left(4\sqrt{\frac{K(t-1)}{\epsilon} \ln \frac{8n^b|\Lambda|}{\delta}} + 2\frac{K}{\epsilon} \ln \frac{8n^b|\Lambda|}{\delta}, t-1\right) \le 16\sqrt{\frac{K(t-1)}{\epsilon} \ln \frac{8n^b|\Lambda|}{\delta}}.$$

In other words,

$$\sum_{\tau=1}^{t-1} \mathbb{E} c^b(\pi_\tau^{\lambda_t}(x)) - \sum_{\tau=1}^{t-1} \mathbb{E} c^b(\pi_\tau^{\lambda}(x)) \le 16\sqrt{\frac{K(t-1)}{\epsilon} \ln \frac{8n^b|\Lambda|}{\delta}}.$$

As the above holds for any λ in Λ, the lemma follows. □

Lemma 6. *For any positive real numbers $a, b > 0$, we have*

$$\min(\sqrt{ab} + b, a) \le 2\sqrt{ab}.$$

Proof. The lemma follows from the straightforward calculations below:

$$\min(\sqrt{ab} + b, a) \le \min(\sqrt{ab}, a) + \min(b, a) \le \sqrt{ab} + \sqrt{ab} = 2\sqrt{ab}.$$ □

F. Proof of Theorem 2

We begin with some notation used in our analysis. Throughout this section, we let $\pi^* = \operatorname{argmin}_{\pi\in\Pi} \mathbb{E}c^s(\pi(x))$, the optimal policy in Π with respect to D^s. Define

$$H_t(\lambda,\alpha,\Delta) = \frac{2\lambda\Delta + 2W_t(\lambda)}{(1-\lambda)+\lambda\alpha}.$$

For policy π, λ in $[0,1]$ and t_e for $e \in \{1,2,\dots,E\}$, define the λ-weighted empirical cost[6] of π as

$$\hat{M}_{\lambda,t_e}(\pi) = \lambda \mathbb{E}_{t_e} \hat{c}(\pi(x)) + (1-\lambda)\mathbb{E}_{S^{\text{tr}}} c^s(\pi(x))$$

and its expectation

$$M_{\lambda,t_e}(\pi) = \lambda \mathbb{E}c^b(\pi(x)) + (1-\lambda)\mathbb{E}c^s(\pi(x))$$

For convenience, we define $\tilde{n}^s = n^s/(E+1)$.

Denote by $(x_{n^b+1}, c^s_{n^b+1}), \dots, (x_{n^b+\tilde{n}^s}, c^s_{n^b+\tilde{n}^s})$ an enumeration of the elements in S^{tr}. Define filtration $\{\mathcal{B}_t\}_{t=0}^{n^b+\tilde{n}^s}$ as follows: $\mathcal{B}_0$ is the trivial σ-algebra, and

$$\mathcal{B}_t = \begin{cases} \sigma((x_{n^b+1}, c^s_{n^b+1}), \dots, (x_{n^b+1}, c^s_{n^b+t})), & t \in \{0,\dots,\tilde{n}^s\}, \\ \sigma(S^{\text{tr}}, (x_1,\hat{c}_1), (x_{t-\tilde{n}^s}, \hat{c}_{t-\tilde{n}^s})), & t \in \left\{\tilde{n}^s+1,\dots,\tilde{n}^s+n^b\right\}. \end{cases}$$

For reader's convenience, we also recall our earlier notation:

$$W_t(\lambda) = 2\sqrt{\left(\tfrac{\lambda^2 K}{t\epsilon} + \tfrac{(1-\lambda)^2(E+1)}{n^s}\right)\ln\tfrac{8E|\Pi|}{\delta}} + \left(\tfrac{\lambda K}{t\epsilon} + \tfrac{(1-\lambda)(E+1)}{n^s}\right)\ln\tfrac{8E|\Pi|}{\delta}.$$

Proof of Theorem 2. Define event I as: for all π in Π,

$$\left|\frac{1}{n^b}\sum_{t=1}^{n^b} \mathbb{E}[c^s(\pi(x_t))|x_t] - \mathbb{E}c^s(\pi(x))\right| \le \sqrt{\frac{\ln\frac{8|\Pi|}{\delta}}{2n^b}}. \tag{13}$$

Intuitively, under this event our regret measure for a policy is close to its expected value. By Hoeffding's inequality and union bound, I happens with probability $1-\frac{\delta}{4}$. Specifically, on event I, as π^* is the policy in Π that minimizes $\mathbb{E}c^s(\pi(x))$, we have that for all π in Π,

$$\mathbb{E}c^s(\pi^*(x)) - \frac{1}{n^b}\sum_{t=1}^{n^b}\mathbb{E}[c^s(\pi(x_t))|x_t] \le \mathbb{E}c^s(\pi(x)) - \frac{1}{n^b}\sum_{t=1}^{n^b}\mathbb{E}[c^s(\pi(x_t))|x_t] \le \sqrt{\frac{\ln\frac{8|\Pi|}{\delta}}{2n^b}}.$$

Therefore,

$$\mathbb{E}c^s(\pi^*(x)) - \min_{\pi\in\Pi}\frac{1}{n^b}\sum_{t=1}^{n^b}\mathbb{E}[c^s(\pi(x_t))|x_t] \le \sqrt{\frac{\ln\frac{8|\Pi|}{\delta}}{2n^b}}. \tag{14}$$

Recall that randomized policy $\pi_t : \mathcal{X} \to \Delta^{K-1}$ is defined as

$$\pi_t(x) = (1-\epsilon)\pi_e^{\lambda_e}(x) + \frac{\epsilon}{K}\mathbb{1}_K, \tag{15}$$

for e in $\{1,2,\dots,E-1\}$, $t \in (t_e, t_{e+1}]$, and $\pi_t(x) = \frac{1}{K}\mathbb{1}_K$ for all t in $(t_0,t_1]$. With a slight abuse of notation, denote by $\mathbb{E}c^s(\pi_t(x)) := \mathbb{E}_{(x,c^s),a\sim\pi_t(x)} c^s(a)$. Observe that for t in $(t_e,t_{e+1}]$, $\mathbb{E}[\mathbb{E}[c^s(a_t)|x_t]|\mathcal{B}_{\tilde{n}^s+t-1}] = \mathbb{E}c^s(\pi_t(x))$. Define event J as:

$$\left|\frac{1}{n^b}\sum_{t=1}^{n^b}\mathbb{E}[c^s(a_t)|x_t] - \frac{1}{n^b}\sum_{t=1}^{n^b}\mathbb{E}c^s(\pi_t(x))\right| \le \sqrt{\frac{\ln\frac{8}{\delta}}{2n^b}}. \tag{16}$$

[6] Note that previously we used a cost of λ per example, whereas now it is λ per source, implying that the per example costs are λ/t_e and $(1-\lambda)/n^s$ after t_e CB examples.

By Azuma's inequality, J happens with probability $1 - \frac{\delta}{4}$.

Denote by E the event that the events E_e, F_e defined in Lemmas 7 and 8 (both given below) and I, J hold simultaneously for all t. By union bound over all E_e, F_e's and I, J, event E happens with probability $1 - \delta$. We henceforth condition on E happening.

Consider e in $\{1, 2, \dots, E-1\}$ and t in $(t_e, t_{e+1}]$. We now upper bound the expected excess cost of using randomized prediction π_t as defined in Equation (15). By the definition of π_t, we get

$$\mathbb{E}c^s(\pi_t(x)) = (1-\epsilon)\mathbb{E}c^s(\pi_e(x)) + \epsilon\frac{1}{K}\sum_{a=1}^{K}\mathbb{E}c^s(a) \leq \mathbb{E}c^s(\pi_e(x)) + \epsilon. \tag{17}$$

Now, combining the above inequality with Lemmas 7 and 8, we have that for all t in $(t_e, t_{e+1}]$,

$$\begin{aligned}
\mathbb{E}c^s(\pi_t(x)) &\leq \min_{\lambda\in\Lambda}\mathbb{E}c^s(\pi_e^\lambda(x)) + \epsilon + \sqrt{\frac{2\ln\frac{8E|\Lambda|}{\delta}}{\tilde{n}^s}} \\
&\leq \mathbb{E}c^s(\pi^*(x)) + \epsilon + \sqrt{\frac{2\ln\frac{8E|\Lambda|}{\delta}}{\tilde{n}^s}} + \min_{\lambda\in\Lambda} H_{t_e}(\lambda, \alpha, \Delta).
\end{aligned}$$

Summing the above inequality over all $t = t_1 + 1, \dots, n^b$, grouping by epoch e, and using the fact that $0 \leq \mathbb{E}c^s(\pi_t(x)) \leq 1$ for $t \leq t_1 = 2$, we have

$$\sum_{t=t_1+1}^{n^b}\mathbb{E}c^s(\pi_t(x)) \leq n^b\mathbb{E}c^s(\pi^*(x)) + 2 + n^b\epsilon + (n^b - 2)\sqrt{\frac{2\ln\frac{8E|\Lambda|}{\delta}}{\tilde{n}^s}} + \sum_{e=1}^{E-1}\sum_{t=t_e+1}^{t_{e+1}}\min_{\lambda\in\Lambda} H_{t_e}(\lambda, \alpha, \Delta).$$

Dividing both sides by n^b and some algebra yields

$$\frac{1}{n^b}\sum_{t=1}^{n^b}\mathbb{E}c^s(\pi_t(x)) \leq \frac{2}{n^b} + \mathbb{E}c^s(\pi^*(x)) + \epsilon + \sqrt{\frac{2\ln\frac{8E|\Lambda|}{\delta}}{\tilde{n}^s}} + \frac{1}{n^b}\sum_{e=1}^{E-1}\sum_{t=t_e+1}^{t_{e+1}}\min_{\lambda\in\Lambda} H_{t_e}(\lambda, \alpha, \Delta).$$

Note that for every t in $[t_e + 1, t_{e+1}]$, as $t \leq t_{e+1} = 2t_e$, $W_t(\lambda) \geq \frac{1}{2}W_{t_e}(\lambda)$, hence $H_t(\lambda, \alpha, \Delta) \geq \frac{1}{2}H_{t_e}(\lambda, \alpha, \Delta)$. Therefore, the right hand side can be further upper bounded by

$$\begin{aligned}
&\frac{2}{n^b} + \mathbb{E}c^s(\pi^*(x)) + \epsilon + \sqrt{\frac{2\ln\frac{8E|\Lambda|}{\delta}}{\tilde{n}^s}} + \frac{2}{n^b}\sum_{e=1}^{E-1}\sum_{t=t_e+1}^{t_{e+1}}\min_{\lambda\in\Lambda} H_t(\lambda, \alpha, \Delta) \\
\leq\; &\frac{2}{n^b} + \mathbb{E}c^s(\pi^*(x)) + \epsilon + \sqrt{\frac{2\ln\frac{8E|\Lambda|}{\delta}}{\tilde{n}^s}} + \min_{\lambda\in\Lambda}\frac{2}{n^b}\sum_{t=1}^{n^b} H_t(\lambda, \alpha, \Delta).
\end{aligned}$$

where the inequality is from that for any set of functions $\{N_t(\cdot)\}_{t=1}^{n^b}$, $\sum_{t=1}^{n^b}\min_{\lambda\in\Lambda} N_t(\lambda) \leq \min_{\lambda\in\Lambda}[\sum_{t=1}^{n^b} N_t(\lambda)]$.

To summarize, we have that

$$\frac{1}{n^b}\sum_{t=1}^{n^b}\mathbb{E}c^s(\pi_t(x)) - \mathbb{E}c^s(\pi^*(x)) \leq \frac{2}{n^b} + \epsilon + \sqrt{\frac{2\ln\frac{8E|\Lambda|}{\delta}}{\tilde{n}^s}} + \min_{\lambda\in\Lambda}\frac{2}{n^b}\sum_{t=1}^{n^b} H_t(\lambda, \alpha, \Delta).$$

Combining with Equations (14) and (16) and using some algebra, we have

$$\frac{1}{n^b}\sum_{t=1}^{n^b}\mathbb{E}[c_t^s(\pi_t(x_t))|x_t] - \min_{\pi\in\Pi}\frac{1}{n^b}\sum_{t=1}^{n^b}\mathbb{E}[c^s(\pi(x_t))|x_t] \leq \epsilon + 3\sqrt{\frac{\ln\frac{8|\Pi|}{\delta}}{n^b}} + \sqrt{\frac{2\ln\frac{8E|\Lambda|}{\delta}}{\tilde{n}^s}} + \min_{\lambda\in\Lambda}\frac{2}{n^b}\sum_{t=1}^{n^b} H_t(\lambda, \alpha, \Delta).$$

The theorem follows from the definition of $\tilde{n}^s$. □

Lemma 7. *For every e, there exists an event E_e with probability $1 - \frac{\delta}{4E}$, on which for all λ in Λ, the excess cost of π_e^λ can be bounded as:*

$$\mathbb{E}c^s(\pi_e^\lambda(x)) - \mathbb{E}c^s(\pi^*(x)) \le H_{t_e}(\lambda, \alpha, \Delta).$$

Proof. We first show the following concentration inequality: with probability $1 - \frac{\delta}{4E}$, for all π in Π,

$$\left|\hat{M}_{\lambda,t_e}(\pi) - M_{\lambda,t_e}(\pi)\right| \le W_{t_e}(\lambda). \tag{18}$$

To show the above statement, in light of the definitions of $\hat{M}, M$ and W, it suffices to show that for every π in Π, with probability $1 - \frac{\delta}{4E|\Pi|}$, we have

$$\begin{aligned} &\left|\sum_{t=1}^{t_e} \frac{\lambda}{t_e}[\hat{c}_t(\pi(x_t)) - \mathbb{E}c(\pi(x))] + \sum_{(x,c^s)\in S} \frac{1-\lambda}{\tilde{n}^s}[c^s(\pi(x)) - \mathbb{E}c^s(\pi(x))]\right| \\ \le\ & 2\sqrt{(\frac{\lambda^2 K}{t_e \epsilon} + \frac{(1-\lambda)^2}{\tilde{n}^s}) \ln \frac{8E|\Pi|}{\delta}} + (\frac{\lambda K}{t_e \epsilon} + \frac{1-\lambda}{\tilde{n}^s}) \ln \frac{8E|\Pi|}{\delta}. \end{aligned} \tag{19}$$

For a fixed π, applying Lemma 2 with $X_i = \frac{1-\lambda}{\tilde{n}^s}(c^s_{n^b+i}(\pi(x_{n^b+i})) - \mathbb{E}c^s(\pi(x)))$ for i in $\{1, \dots, \tilde{n}^s\}$, $X_i = \frac{\lambda}{t_e}(\hat{c}_{i-\tilde{n}^s}(\pi(x_{i-\tilde{n}^s})) - \mathbb{E}c^b(\pi(x)))$ for i in $\{\tilde{n}^s + 1, \dots, \tilde{n}^s + t_e\}$, and $M = \frac{1-\lambda}{\tilde{n}^s} + \frac{\lambda K}{t_e \epsilon}$, and note that $|X_i| \le M$ almost surely for all i, $\mathbb{E}[X_i^2|\mathcal{B}_{i-1}] \le (\frac{1-\lambda}{\tilde{n}^s})^2$ for i in $\{1, \dots, \tilde{n}^s\}$, and $\mathbb{E}[X_i^2|\mathcal{B}_{i-1}] \le (\frac{\lambda}{t_e})^2 \mathbb{E}[\frac{1}{p_{i-\tilde{n}^s, \pi(x_{i-\tilde{n}^s})}}|\mathcal{B}_{i-1}] \le \frac{\lambda^2 K}{t_e^2 \epsilon}$ for i in $\{\tilde{n}^s + 1, \dots, \tilde{n}^s + t_e\}$, we get that Equation (19) holds with probability $1 - \frac{\delta}{4E|\Pi|}$.

Therefore, by union bound over all π in Π Equation (18) holds for all λ simultaneously with probability $1 - \frac{\delta}{4E}$. As π_e^λ minimizes $\hat{M}_{\lambda,t}(\pi)$ over Π, we have that $\hat{M}_{\lambda,t}(\pi_e^\lambda) \le \hat{M}_{\lambda,t}(\pi^*)$. Combining this fact with Equation (18) on π_e^λ and π^*, we get

$$M_{\lambda,t_e}(\pi_e^\lambda) - M_{\lambda,t_e}(\pi^*) \le 2W_{t_e}(\lambda).$$

Observe that Lemma 1 and the definition of M implies that the left hand side of the above equation is at least $(\lambda\alpha + (1-\lambda))(\mathbb{E}c^s(\pi_e^\lambda(x)) - \mathbb{E}c^s(\pi^*(x))) - 2\lambda\Delta$. The lemma statement follows by straightforward algebra and the definition of $H_t(\cdot, \cdot, \cdot)$. ∎

Lemma 8. *For every e, there exists an event F_e with probability $1 - \frac{\delta}{4E}$, on which the policy $\pi_e^{\lambda_e}$ satisfies that*

$$\mathbb{E}c^s(\pi_e^{\lambda_e}) - \min_{\lambda\in\Lambda} \mathbb{E}c^s(\pi_e^\lambda(x)) \le \sqrt{\frac{2\ln\frac{8E|\Lambda|}{\delta}}{\tilde{n}^s}}.$$

Proof. Given any λ in Λ, as S_e^{val} is a sample independent of the π_e^λ, we have by Hoeffding's inequality that with probability $1 - \frac{\delta}{4E|\Lambda|}$,

$$\left|\mathbb{E}_{S_e^{\text{val}}} c^s(\pi_e^\lambda(x)) - \mathbb{E}c^s(\pi_e^\lambda(x))\right| \le \sqrt{\frac{\ln\frac{8E|\Lambda|}{\delta}}{2\tilde{n}^s}}. \tag{20}$$

By union bound, with probability $1 - \frac{\delta}{4}$, Equation (20) holds for all λ in Λ simultaneously.

Observe that by the optimality of λ_e, for all λ in Λ, we have that

$$\mathbb{E}_{S_e^{\text{val}}} c^s(\pi_e^{\lambda_e}(x)) \le \mathbb{E}_{S_e^{\text{val}}} c^s(\pi_e^\lambda(x)).$$

Combining with Equation (20) on λ and λ_e, we get

$$\mathbb{E}c^s(\pi_e^{\lambda_e}(x)) \le \mathbb{E}c^s(\pi_e^{\lambda}(x)) + \sqrt{\frac{2\ln\frac{8E|\Lambda|}{\delta}}{\tilde{n}^s}}.$$

The lemma follows as the above inequality holds for every λ in Λ. □

G. Approximate optimality of $\Lambda = \{0, 1\}$

In this section, we show Proposition 2, which justifies that using $\Lambda = \{0, 1\}$ achieves a near-optimal regret bound in Theorem 1. Recall that

$$V_t(\lambda) = 2\sqrt{\left(\lambda^2\frac{Kt}{\epsilon} + (1-\lambda)^2 n^s\right)\ln\frac{8n^b|\Pi|}{\delta}} + \left(\frac{\lambda K}{\epsilon} + (1-\lambda)\right)\ln\frac{8n^b|\Pi|}{\delta},$$
$$G_t(\lambda, \alpha, \Delta) = \frac{2(1-\lambda)n^s\Delta + 2V_t(\lambda)}{\lambda t + (1-\lambda)n^s\alpha}.$$

Proposition 2.

$$\min_{\lambda\in\{0,1\}} G_t(\lambda, \alpha, \Delta) \le \sqrt{2}\min_{\lambda\in[0,1]} G_t(\lambda, \alpha, \Delta).$$

Proof. For any λ in $[0, 1]$, we give a lower bound on $V_t(\lambda)$. Note that by the fact that $\sqrt{a+b} \ge \sqrt{\frac{1}{2}}(\sqrt{a} + \sqrt{b})$, we have that

$$V_t(\lambda) \ge \frac{1}{\sqrt{2}}[(1-\lambda)V_t(0) + \lambda V_t(1)]. \tag{21}$$

Therefore,

$$\begin{aligned}
& G_t(\lambda, \alpha, \Delta) \\
\ge\ & \frac{2(1-\lambda)n^s\Delta + 2\cdot\sqrt{\frac{1}{2}}[(1-\lambda)V_t(0) + \lambda V_t(1)]}{\lambda t + (1-\lambda)n^s\alpha} \\
\ge\ & \frac{1}{\sqrt{2}}\frac{2(1-\lambda)n^s\Delta + 2[(1-\lambda)V_t(0) + \lambda V_t(1)]}{\lambda t + (1-\lambda)n^s\alpha} \\
\ge\ & \frac{1}{\sqrt{2}}\min(\frac{2n^s\Delta + 2V_t(0)}{n^s\alpha}, \frac{2V_t(1)}{t}) \\
=\ & \frac{1}{\sqrt{2}}\min_{\lambda\in\{0,1\}} G_t(\lambda, \alpha, \Delta),
\end{aligned}$$

where the first inequality is from Equation (21), the second inequality is by algebra, the third inequality is by the quasi-concavity of the LHS with respect to λ, and the fact that the minimum of a quasi-concave function over a convex set is attained at its boundary. As the above holds for any λ in $[0, 1]$, the proposition follows. □

H. Combining Two Sources in Supervised Learning

Proposition 3. *For every policy class Π of VC dimension d and $\Delta \in [0, \frac{1}{4}]$, $m, n \ge 32d$, for any algorithm that outputs a policy $\hat{\pi}$ based on n examples from D^1 and m examples from D^2, there exists a pair of distributions (D^1, D^2) such that D^2 is $(1, \Delta)$-similar to D^1, and*

$$\mathbb{E}[\mathbb{E}_{D^1}c(\hat{\pi}(x)) - \min_{\pi\in\Pi}\mathbb{E}_{D^1}c(\pi(x))] \ge \frac{1}{16}\min\left(\sqrt{d/m} + 8\Delta, \sqrt{d/n}\right)$$

where the outer expectation is over draws from D^1 and D^2, and the algorithm's randomness.

The lower bound proof follows a similar strategy as in the classical classification setting with slight modifications. In order to prove the bound we make use of Assouad's Lemma. The statement below follows exactly from Yu (Yu, 1997).

Theorem 3 (Assouad's Lemma). *Let $d \geq 1$ be an integer and let $\mathcal{F}_d = \{P_\tau \mid \tau \in \{-1,+1\}^d\}$ be a class of 2^d probability measures index by binary strings of length d. Write $\tau \sim \tau^{'}$ if τ and $\tau^{'}$ differ in only one coordinate, and write $\tau \sim_j \tau^{'}$ when that coordinate is the j^{th}. Suppose that there are d psuedo-distances on $\mathcal{D}$ such that for any $x, y \in \mathcal{D}$*

$$\rho(x,y) = \sum_{j=1}^{d} \rho_j(x,y),$$

and further that, if $\tau \sim_j \tau^{'}$,

$$\rho_j(\theta(P_\tau), \theta(P_{\tau'})) \geq \alpha.$$

Then for any estimator $\widehat{\theta}$,

$$\max_\tau \mathbb{E}_\tau \rho(\widehat{\theta}, \theta(P_\tau)) \geq d \cdot \frac{\alpha}{2} \min(\|P_\tau \wedge P_{\tau'}\| \mid \tau \sim \tau^{'}\}.$$

Proof of Proposition 3. Since the VC dimension of Π is d, there exists a set $A = \{x_1, \ldots, x_d\}$ such that for any binary sequence $\tau \in \{-1,+1\}^d$, there exists some function π in Π such that for all $i \in \{1, \ldots, d\}$, $\pi(x_i) = \tau_i$. We now define two distributions for a binary classification problem, but express them as joint distributions over (x, c) in order to be consistent with the rest of the paper with the understanding that c will have a zero-one cost structure (where $c = (c(-1), c(+1))$ represent the costs of predicting labels -1 and $+1$). The two distributions D_τ^1 and D_τ^2 are each uniform over x_i, and the conditional distributions over costs are given by

$$D_\tau^1((1,0)|x = x_i) = \frac{1}{2} + \tau_i\epsilon$$
$$D_\tau^1((0,1)|x = x_i) = \frac{1}{2} - \tau_i\epsilon, \quad \text{and}$$

$$D_\tau^2((1,0)|x = x_i) = \frac{1}{2} + \tau_i\epsilon - \tau_i\beta$$
$$D_\tau^2((0,1)|x = x_i) = \frac{1}{2} - \tau_i\epsilon + \tau_i\beta.$$

Where $\epsilon \in [0, \frac{1}{2}]$, $\beta \in [0, \Delta]$ are free parameters to be determined later. Clearly D_τ^2 is $(1, \Delta)$-similar to D_τ^1. This can be seen by taking $c^+ = \mathbb{E}_{D^1}[c|x]$ and $c^- = \mathbb{E}_{D^2}[c|x] - \mathbb{E}_{D^1}[c|x]$, and observing that $|\mathbb{E}c^-(\pi(x))| \leq \beta \leq \Delta$. Define P_τ as the product distribution of n copies of D_τ^1 and m copies of D_τ^2, which is the joint distribution of the input examples to the algorithm. In addition, define $\theta(P_\tau) = \tau$, and $\rho_j(\tau, \tau') = I(\tau_j \neq \tau'_j)$. Therefore, $\rho(\tau, \tau') = \sum_{j=1}^d \rho_j(\tau, \tau') = \sum_{j=1}^d I(\tau_j \neq \tau'_j)$ is the Hamming distance between τ and τ'. Suppose the algorithm returns a policy $\hat{\pi}$, there exists a binary sequence $\hat{\tau}$ such that $\hat{\pi}(x_i) = \hat{\tau}_i$. Observe that for any τ, by the definition of D_τ^1,

$$\mathbb{E}_{D_\tau^1} c(\hat{\pi}(x)) - \min_{\pi \in \Pi} \mathbb{E}_{D_\tau^1} c(\pi(x)) = \frac{2\epsilon}{d} \rho(\hat{\tau}, \tau).$$

By Assouad's Lemma (Theorem 3) with the ρ defined above and $\alpha = 1$, we have that there exists some τ in $\{-1,+1\}^d$, such that

$$\mathbb{E}\rho(\hat{\tau}, \tau) \geq \epsilon \min(\|P_\tau \wedge P_{\tau'}\|_1 \mid \tau \sim \tau^{'})$$

This immediately implies that, for any algorithm that returns $\hat{\pi}$, there exists some τ such that

$$\mathbb{E}_{P_\tau}[c(\hat{\pi}(x)) - \min_{\pi \in \Pi} \mathbb{E}_{D_\tau^1} c(\pi(x))] \geq \epsilon \min(\|P_\tau \wedge P_{\tau'}\|_1 \mid \tau \sim \tau^{'}).$$

What remains is to bound $\|P_\tau \wedge P_{\tau'}\|_1$ for any binary sequences differing in one coordinate. Recall that $\|P_\tau \wedge P_{\tau'}\| = 1 - \frac{1}{2}\|P_\tau - P_{\tau'}\|_1$. We will bound $\frac{1}{2}\|P_\tau - P_{\tau'}\|_1 \leq H(P_\tau, P_{\tau'})$ using the Hellinger distance $H^2(P_\tau, P_{\tau'}) = \frac{1}{2}\sum_z (\sqrt{P_\tau(z)} - \sqrt{P_{\tau'}(z)})^2$. We recall that P_τ is in fact the product distribution over n copies of D_τ^1 and m copies of D_τ^2. For product measures

$$H^2(P_\tau, P_{\tau'}) \leq \sum_{i=1}^{n} (D_\tau^1, D_{\tau'}^1) + \sum_{i=1}^{m} (D_\tau^2, D_{\tau'}^2)$$

we need to bound the Hellinger distance for the biased and unbiased distributions.

Bounding the Hellinger Distance We have

$$\begin{aligned} H^2(D_\tau^1, D_{\tau'}^1) &= \frac{1}{2}\sum_{i=1}^{d}\sum_{c\in\{(0,1),(1,0)\}}(\sqrt{D_\tau^1(x_i,c)} - \sqrt{D_{\tau'}^1(x_i,c)})^2 \\ &= \frac{1}{2d}\sum_{c\in\{(0,1),(1,0)\}}(\sqrt{D_\tau^1(c|x_i)} - \sqrt{D_{\tau'}^1(c|x_i)})^2 \\ &= \frac{1}{d}H^2(B(\frac{1}{2}+\epsilon), B(\frac{1}{2}-\epsilon)) \\ &\le \frac{8}{d}\epsilon^2, \end{aligned}$$

where the first equality is by the definiton of H^2; the second inequality is from that there is exactly one i such that $\sum_{c\in\{(0,1),(1,0)\}}(\sqrt{D_\tau^1(x_i,c)} - \sqrt{D_{\tau'}^1(x_i,c)})^2$ is nonzero, as $\tau \sim \tau'$; in the right hand side of third equality, $B(p)$ is the Bernoulli distribution with mean parameter p. Similarly,

$$H^2(D_\tau^2, D_{\tau'}^2) \le \frac{8}{d}(\epsilon-\beta)^2$$

Hence,

$$H^2(P_\tau, P_{\tau'}) \le \frac{8}{d}\left(n\epsilon^2 + m(\epsilon-\beta)^2\right)$$

Therefore, we have

$$\max_\tau \mathbb{E}_{P_\tau}[c(\hat{\pi}(x)) - \min_{\pi\in\Pi}\mathbb{E}_{D_\tau^1}c(\pi(x))] \ge \epsilon\left[1 - \sqrt{\frac{8}{d}\left(n\epsilon^2 + m(\epsilon-\beta)^2\right)}\right] \tag{22}$$

We now consider two separate cases regarding the settings of m, n, d and Δ.

Case 1: $\sqrt{\frac{d}{n}} \le \sqrt{\frac{d}{m}} + 8\Delta$. In this case, we let $\epsilon = \frac{1}{8}\sqrt{\frac{d}{n}}$ and $\beta = \frac{1}{8}\max(0, \sqrt{\frac{d}{n}} - \sqrt{\frac{d}{m}}) \in [0, \Delta]$. This gives that the right hand side of Equation (22) is at least $\frac{1}{8}\sqrt{\frac{d}{n}}\cdot\frac{1}{2} = \frac{1}{16}\sqrt{\frac{d}{n}}$.

Case 2: $\sqrt{\frac{d}{n}} > \sqrt{\frac{d}{m}} + 8\Delta$. In this case, we let $\epsilon = \frac{1}{8}\sqrt{\frac{d}{m}} + \Delta$ and $\beta = \Delta$. This gives that the right hand side of Equation (22) is at least $(\frac{1}{8}\sqrt{\frac{d}{m}} + \Delta)\cdot\frac{1}{2} = \frac{1}{16}(\sqrt{\frac{d}{m}} + 8\Delta)$.

In summary, for any choice of m, n, d, Δ with $m, n \ge 32d$ and $\Delta \in [0, \frac{1}{4}]$, we can find a pair of distributions (D_τ^1, D_τ^2) such that

$$\mathbb{E}_{P_\tau}[c(\hat{\pi}(x)) - \min_{\pi\in\Pi}\mathbb{E}_{D_\tau^1}c(\pi(x))] \ge \frac{1}{16}\min(\sqrt{\frac{d}{n}}, \sqrt{\frac{d}{m}} + 8\Delta).$$

The proposition follows. □

I. Additional Experimental Results

We give a collection of cumulative distribution functions (CDFs) of our algorithms evaluated against different exploration parameters ϵ, noise settings and warm start ratios below:

1. In Figures 4 to 9, we present CDFs where all CB algorithms use ϵ-greedy with parameter $\epsilon = 0.0125$.
2. In Figures 10 to 15, we present CDFs where all CB algorithms use ϵ-greedy with parameter $\epsilon = 0.00625$.
3. In Figures 16 to 21, we present CDFs where all CB algorithms use ϵ-greedy with parameter $\epsilon = 0.1$.

The general trends in this more detailed comparison are similar to those observed in Section 5. For less noisy and small warm-start ratios, SUP-ONLY is particularly difficult as a baseline since it performs no exploration. With extreme noise, BANDIT-ONLY is the best as the supervised examples are misleading. AWESOMEBANDITS competes well on both the extremes, while outperforming all methods in several regimes. Importantly, AWESOMEBANDITS always beats the other methods that attempt to leverage both sources of data, and prevents the significant performance hit from relying on the wrong data source in either of the two extreme cases.

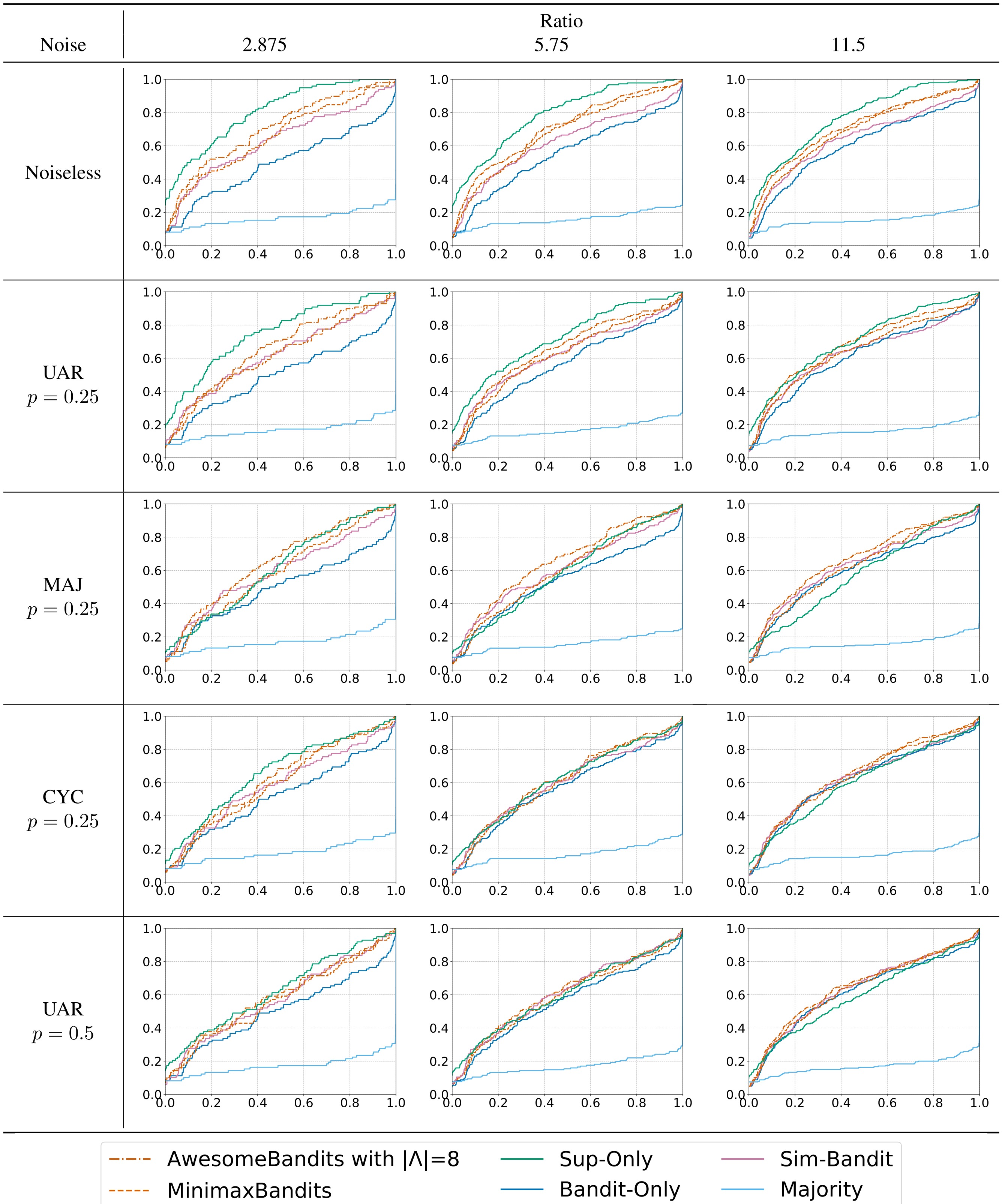


Figure 4: Cumulative distribution functions (CDFs) for all evaluated algorithms, against different noise settings and warm start ratios in $\{2.875, 5.75, 11.5\}$. All CB algorithms use ϵ-greedy with $\epsilon = 0.0125$. In each of the above plots, the x axis represents scores, while the y axis represents the CDF values.

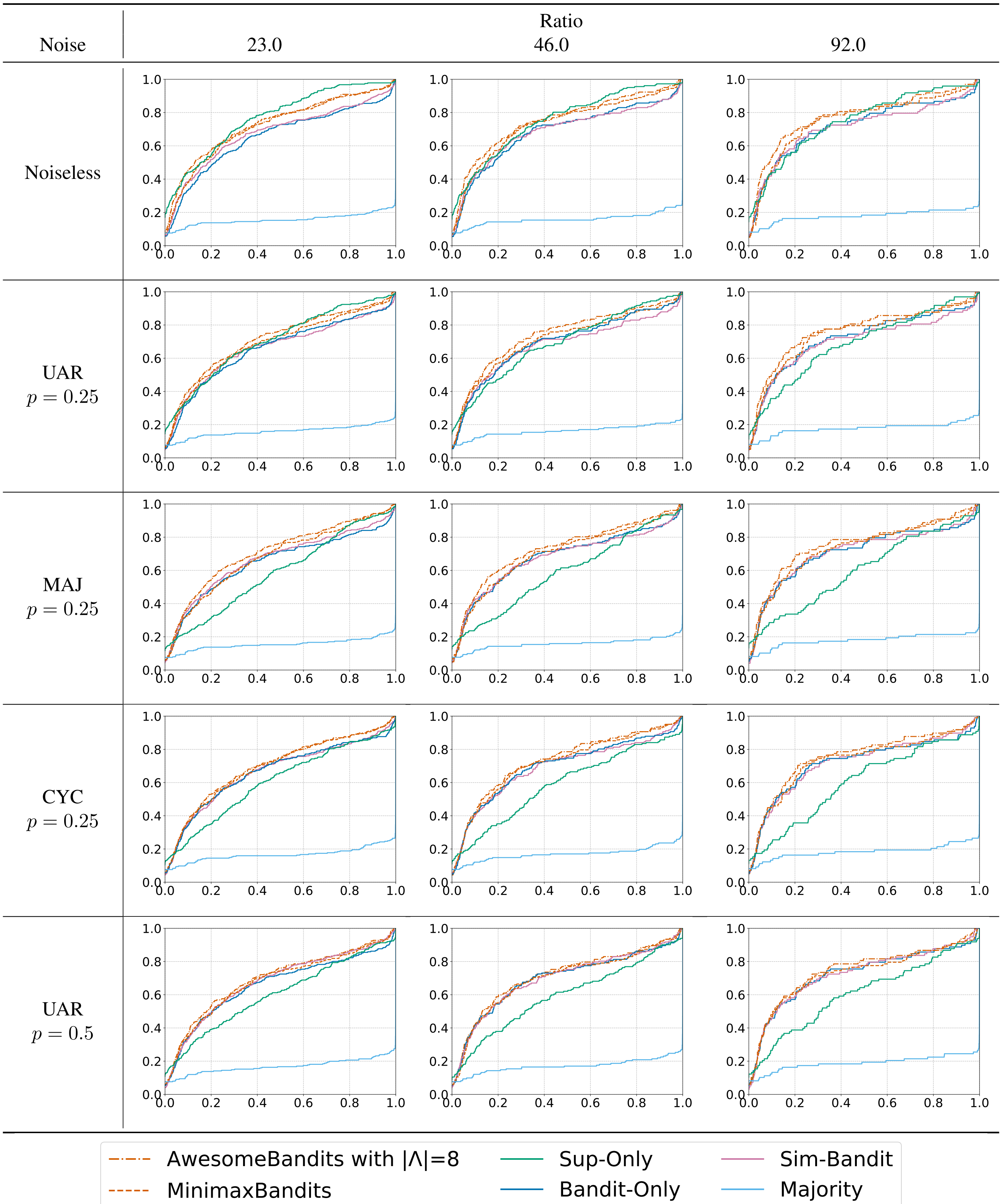
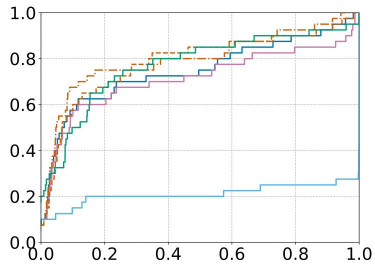
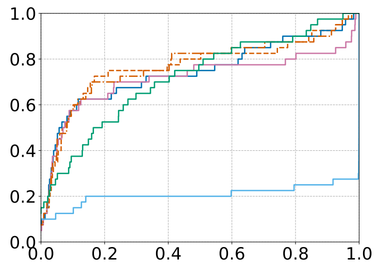
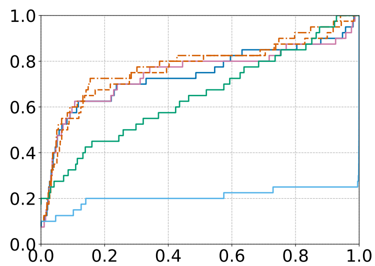
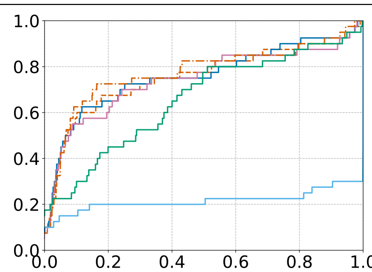
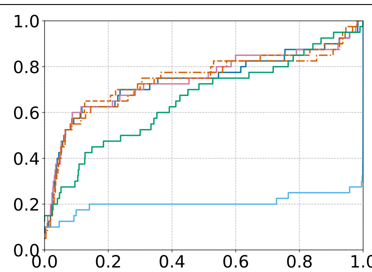


Figure 5: Cumulative distribution functions (CDFs) for all evaluated algorithms, against different noise settings and warm start ratios in $\{23.0, 46.0, 92.0\}$. All CB algorithms use ϵ-greedy with $\epsilon = 0.0125$. In each of the above plots, the x axis represents scores, while the y axis represents the CDF values.

Noise	Ratio 184.0
Noiseless	
UAR $p = 0.25$	
MAJ $p = 0.25$	
CYC $p = 0.25$	
UAR $p = 0.5$	

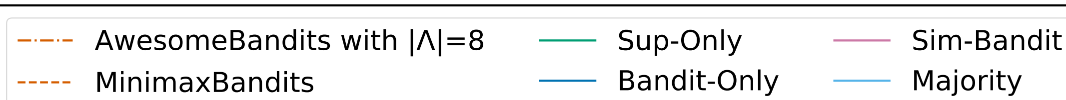


Figure 6: Cumulative distribution functions (CDFs) for all evaluated algorithms, against different noise settings and warm start ratios in $\{184.0\}$. All CB algorithms use ϵ-greedy with $\epsilon = 0.0125$. In each of the above plots, the x axis represents scores, while the y axis represents the CDF values.

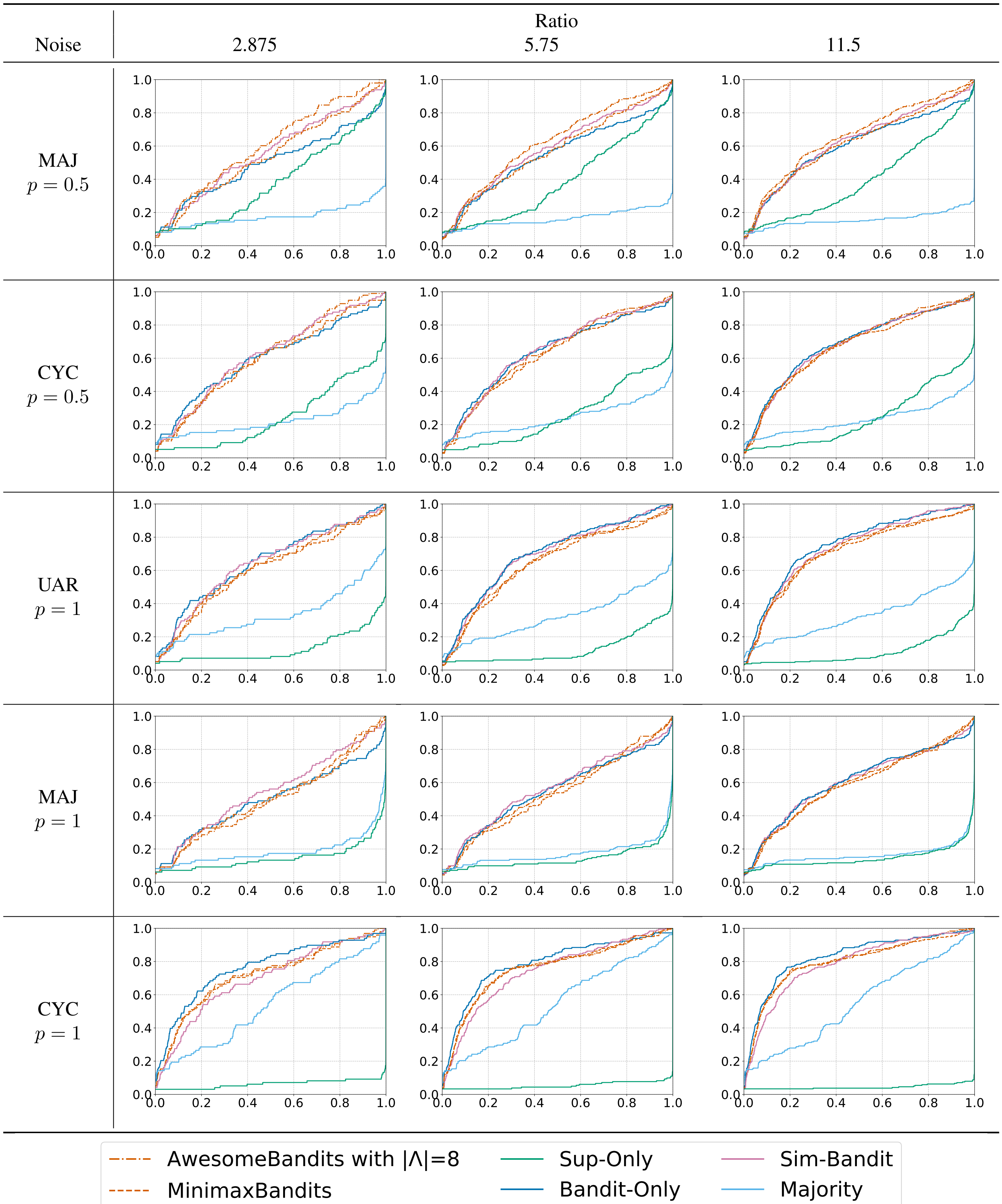


Figure 7: Cumulative distribution functions (CDFs) for all evaluated algorithms, against different noise settings and warm start ratios in $\{2.875, 5.75, 11.5\}$. All CB algorithms use ϵ-greedy with $\epsilon = 0.0125$. In each of the above plots, the x axis represents scores, while the y axis represents the CDF values.

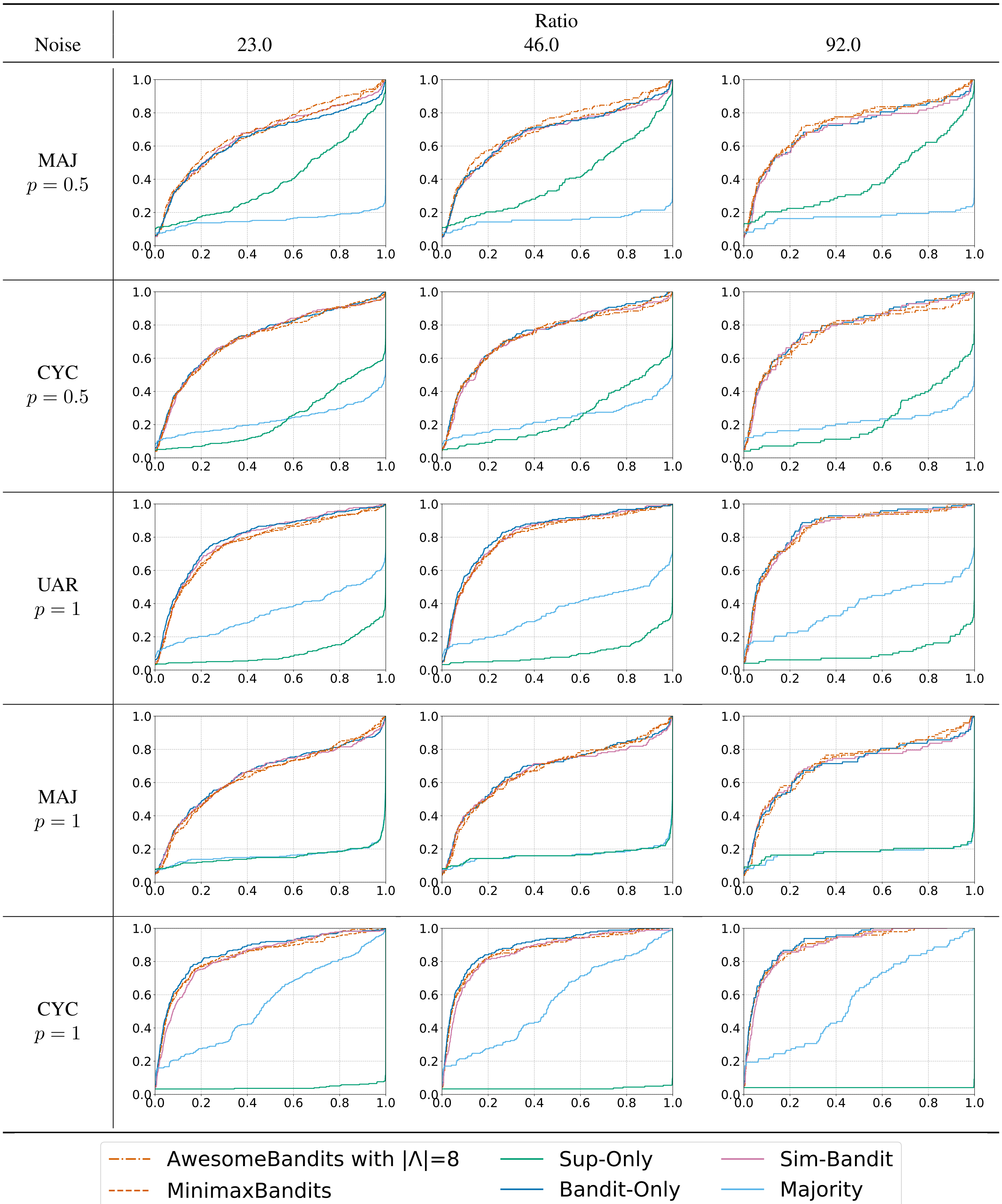
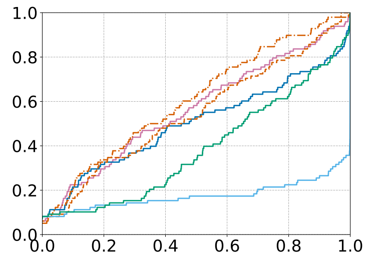
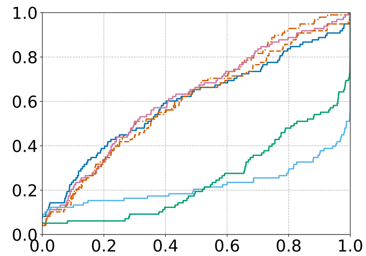
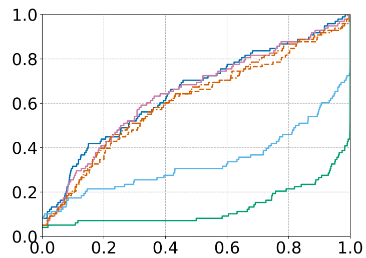
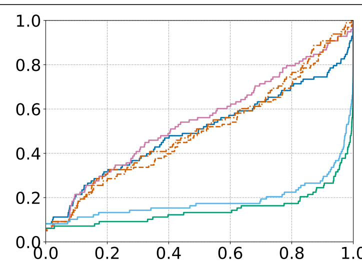
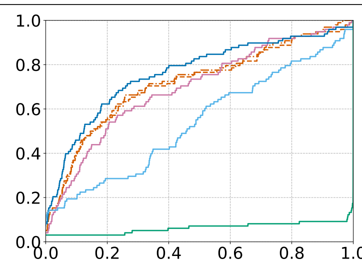


Figure 8: Cumulative distribution functions (CDFs) for all evaluated algorithms, against different noise settings and warm start ratios in $\{23.0, 46.0, 92.0\}$. All CB algorithms use ϵ-greedy with $\epsilon = 0.0125$. In each of the above plots, the x axis represents scores, while the y axis represents the CDF values.

Noise	Ratio 184.0
MAJ $p = 0.5$	
CYC $p = 0.5$	
UAR $p = 1$	
MAJ $p = 1$	
CYC $p = 1$	

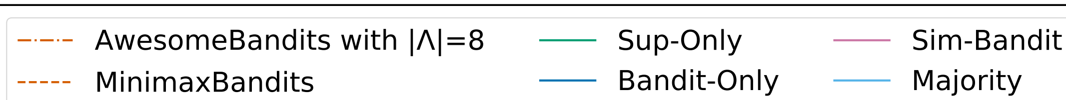


Figure 9: Cumulative distribution functions (CDFs) for all evaluated algorithms, against different noise settings and warm start ratios in $\{184.0\}$. All CB algorithms use ϵ-greedy with $\epsilon = 0.0125$. In each of the above plots, the x axis represents scores, while the y axis represents the CDF values.

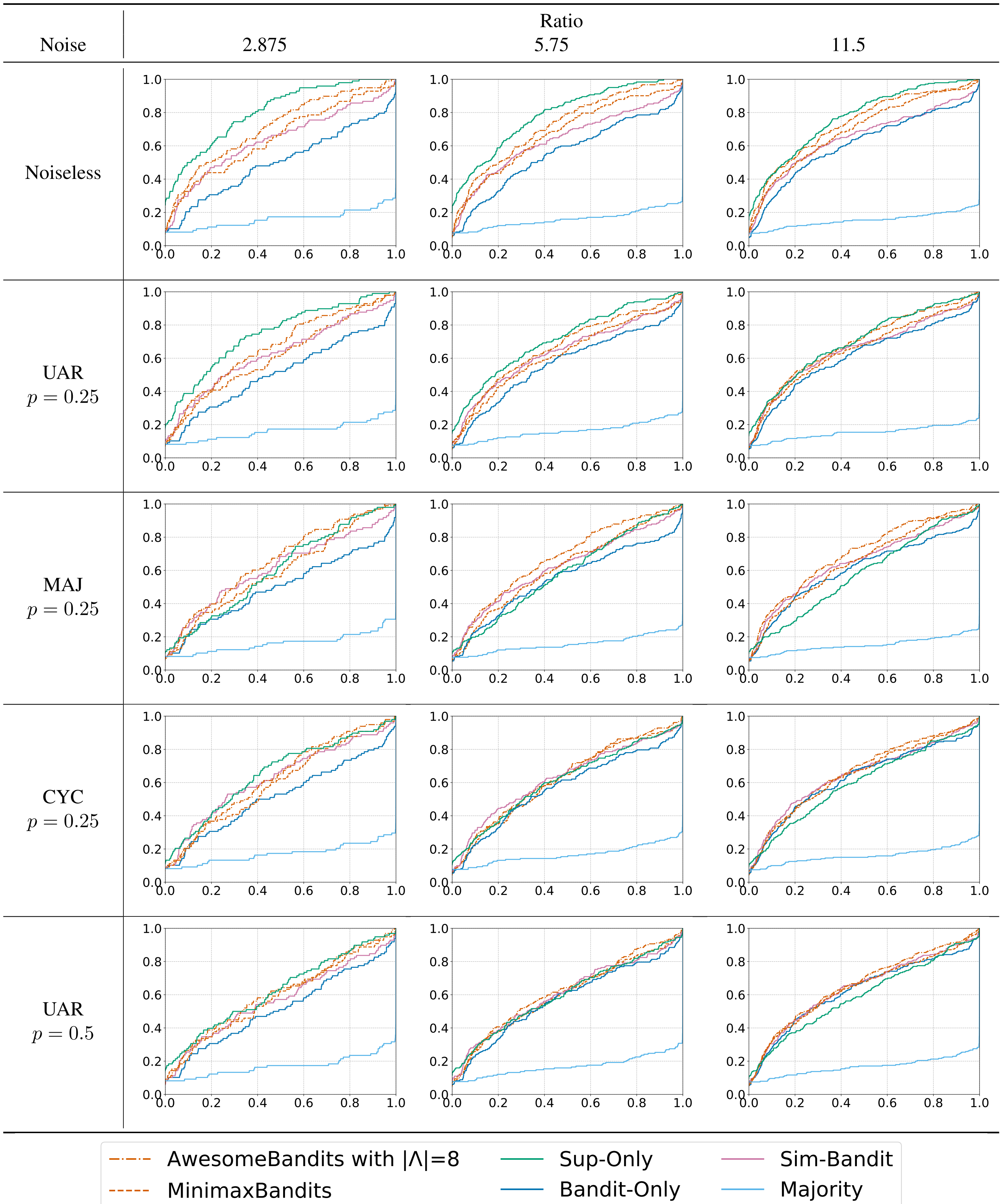


Figure 10: Cumulative distribution functions (CDFs) for all evaluated algorithms, against different noise settings and warm start ratios in $\{2.875, 5.75, 11.5\}$. All CB algorithms use ϵ-greedy with $\epsilon = 0.00625$. In each of the above plots, the x axis represents scores, while the y axis represents the CDF values.

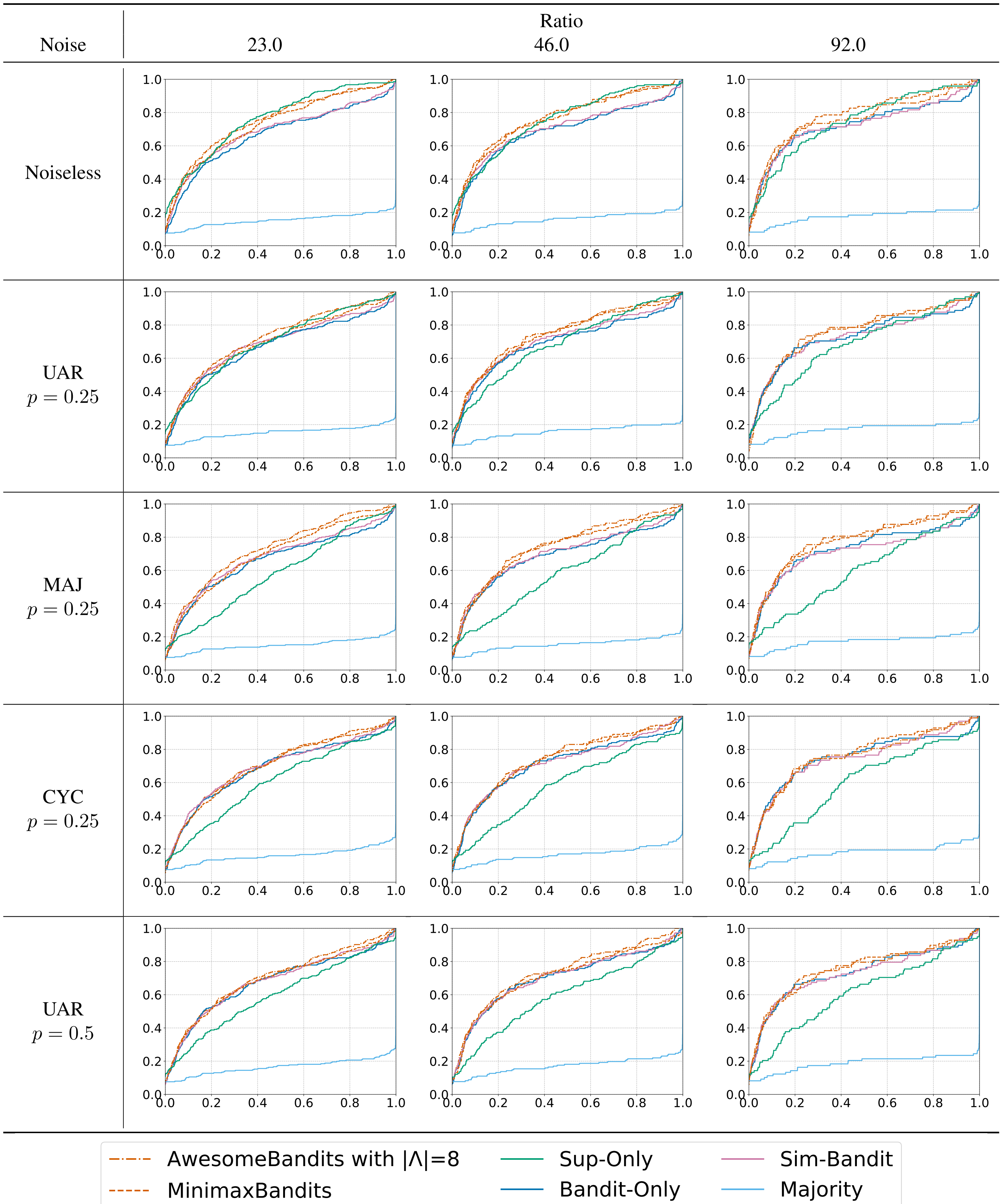


Figure 11: Cumulative distribution functions (CDFs) for all evaluated algorithms, against different noise settings and warm start ratios in $\{23.0, 46.0, 92.0\}$. All CB algorithms use ϵ-greedy with $\epsilon = 0.00625$. In each of the above plots, the x axis represents scores, while the y axis represents the CDF values.

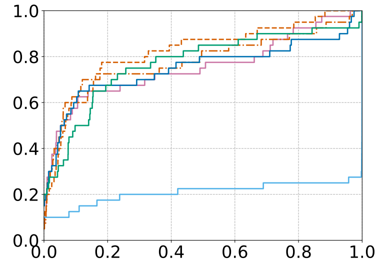
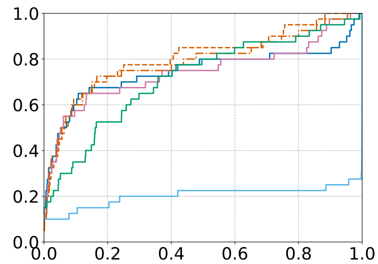
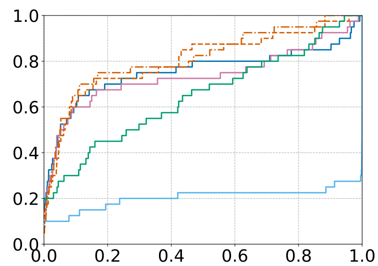
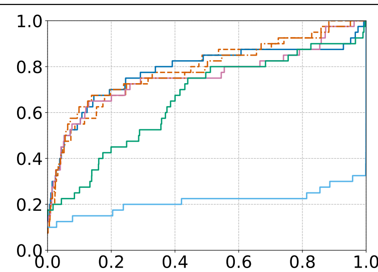
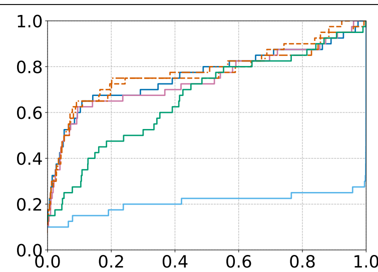
Noise	Ratio 184.0
Noiseless	
UAR $p = 0.25$	
MAJ $p = 0.25$	
CYC $p = 0.25$	
UAR $p = 0.5$	

Figure 12: Cumulative distribution functions (CDFs) for all evaluated algorithms, against different noise settings and warm start ratios in $\{184.0\}$. All CB algorithms use ϵ-greedy with $\epsilon = 0.00625$. In each of the above plots, the x axis represents scores, while the y axis represents the CDF values.

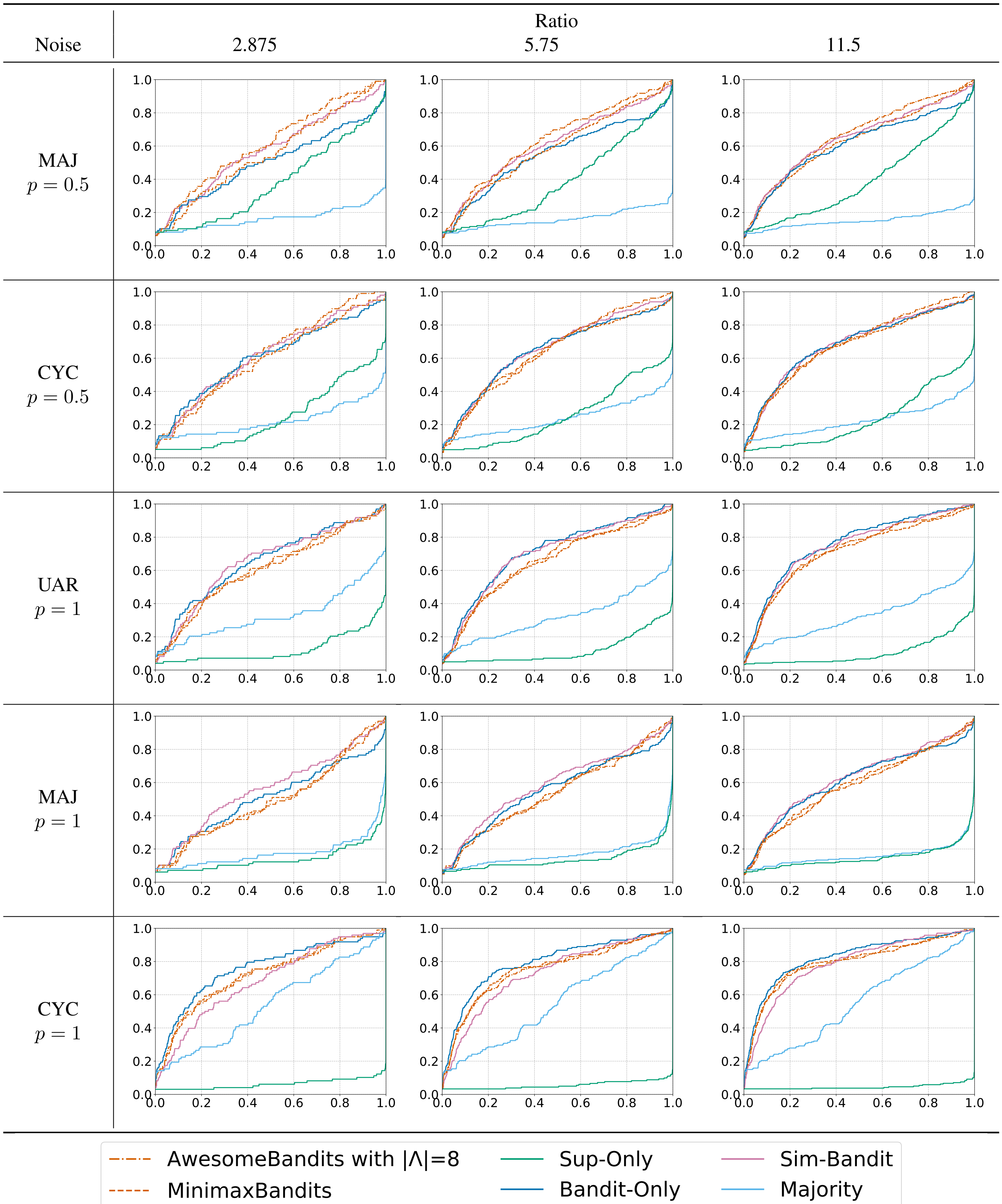


Figure 13: Cumulative distribution functions (CDFs) for all evaluated algorithms, against different noise settings and warm start ratios in $\{2.875, 5.75, 11.5\}$. All CB algorithms use ϵ-greedy with $\epsilon = 0.00625$. In each of the above plots, the x axis represents scores, while the y axis represents the CDF values.

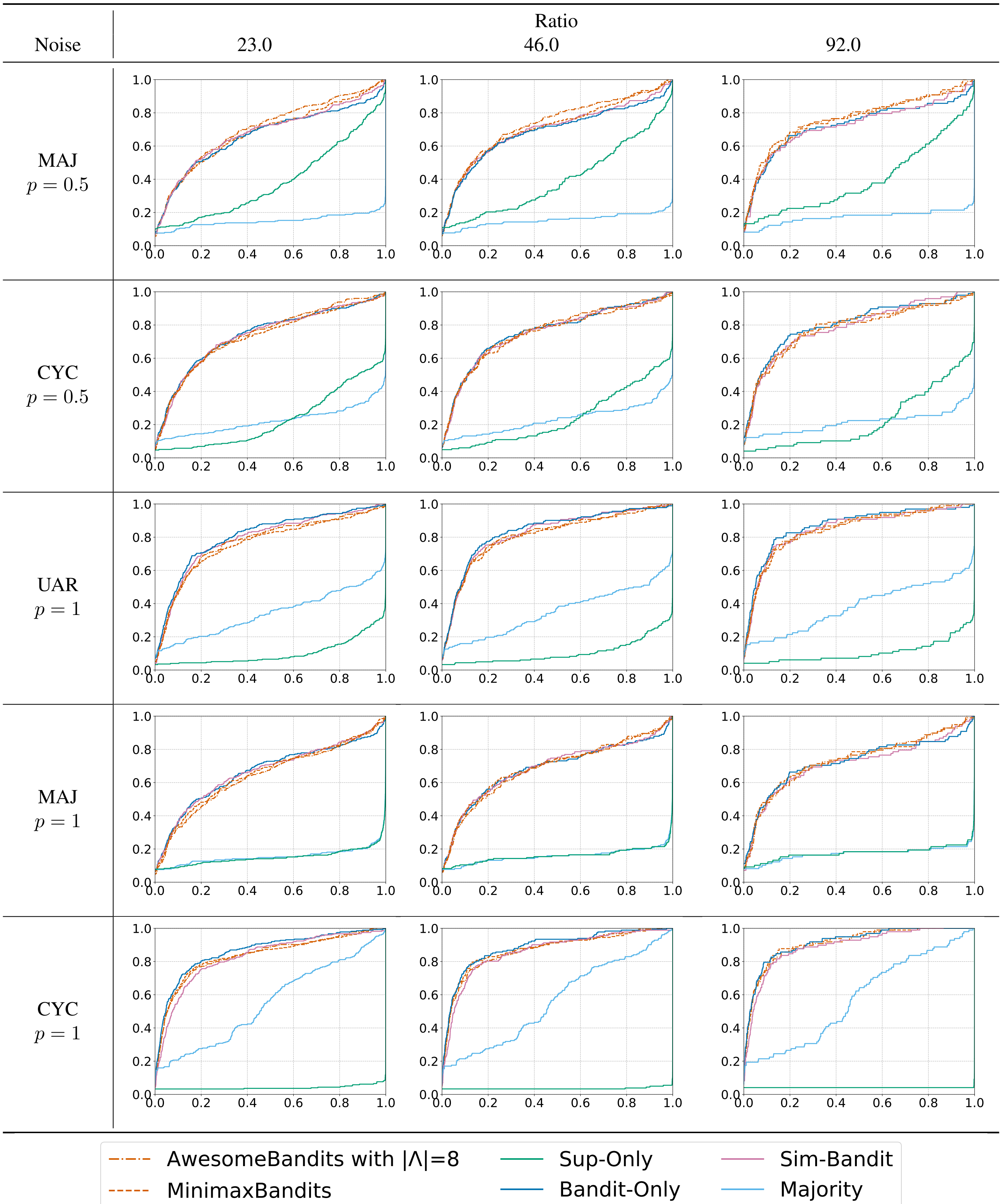
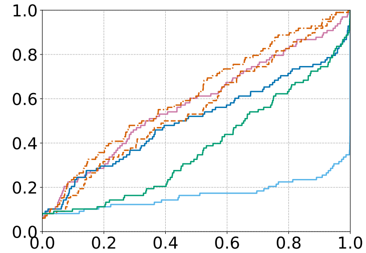
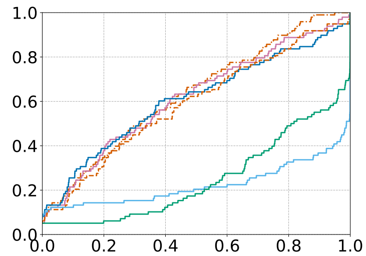
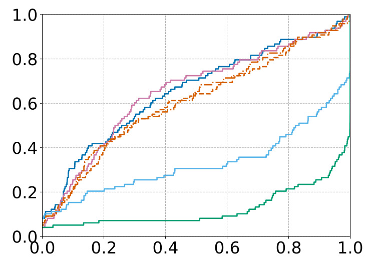
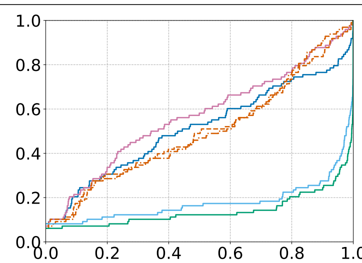
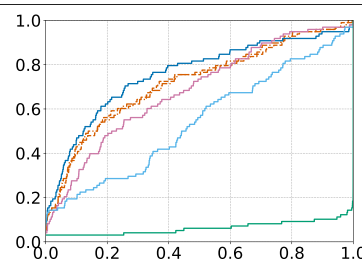


Figure 14: Cumulative distribution functions (CDFs) for all evaluated algorithms, against different noise settings and warm start ratios in $\{23.0, 46.0, 92.0\}$. All CB algorithms use ϵ-greedy with $\epsilon = 0.00625$. In each of the above plots, the x axis represents scores, while the y axis represents the CDF values.

Noise	Ratio 184.0
MAJ $p = 0.5$	
CYC $p = 0.5$	
UAR $p = 1$	
MAJ $p = 1$	
CYC $p = 1$	

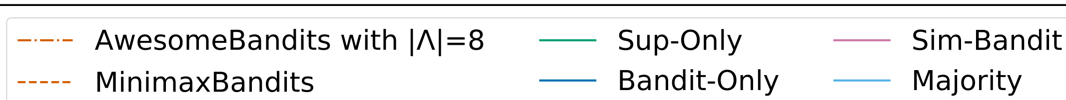


Figure 15: Cumulative distribution functions (CDFs) for all evaluated algorithms, against different noise settings and warm start ratios in $\{184.0\}$. All CB algorithms use ϵ-greedy with $\epsilon = 0.00625$. In each of the above plots, the x axis represents scores, while the y axis represents the CDF values.

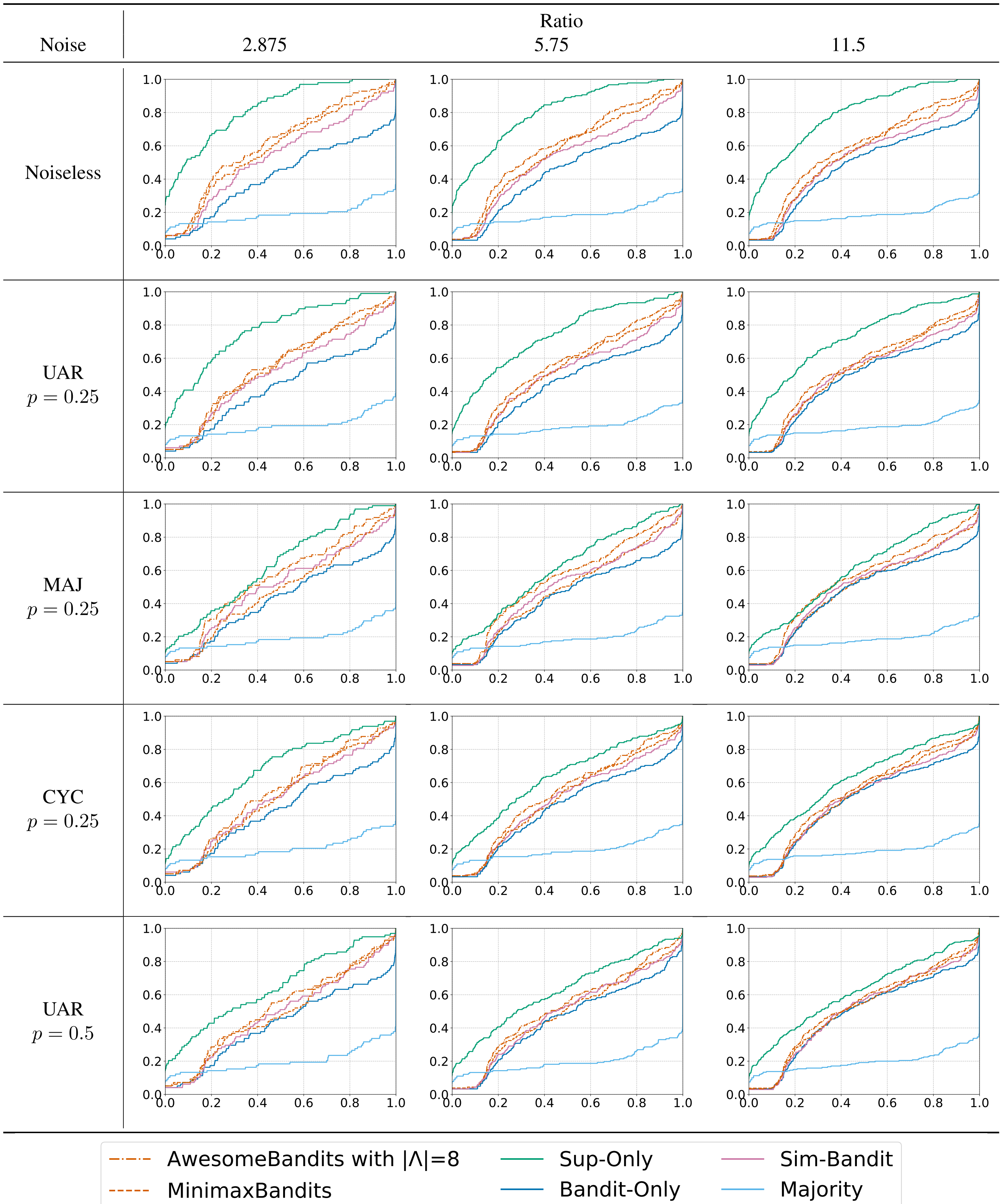


Figure 16: Cumulative distribution functions (CDFs) for all evaluated algorithms, against different noise settings and warm start ratios in $\{2.875, 5.75, 11.5\}$. All CB algorithms use ϵ-greedy with $\epsilon = 0.1$. In each of the above plots, the x axis represents scores, while the y axis represents the CDF values.

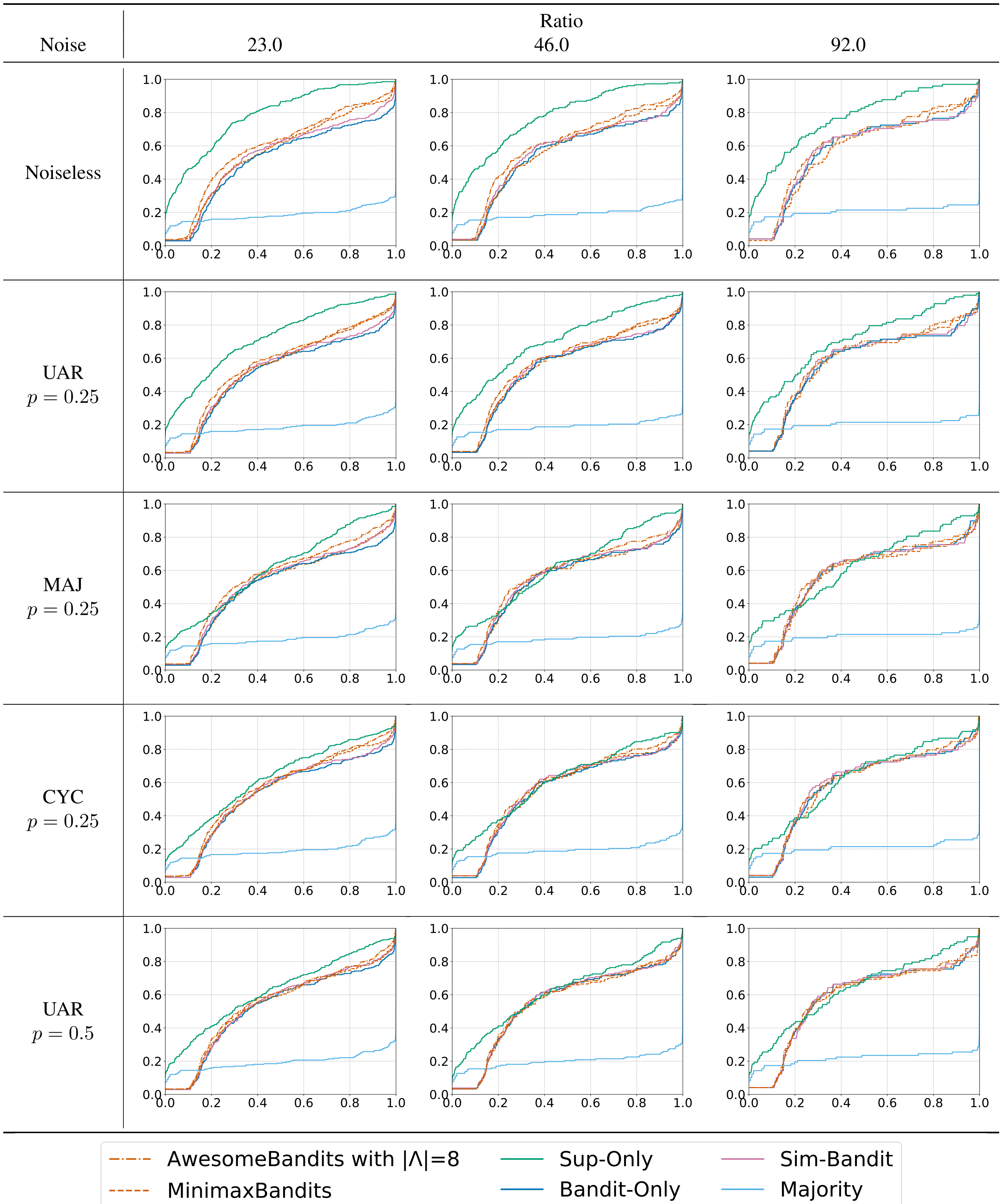


Figure 17: Cumulative distribution functions (CDFs) for all evaluated algorithms, against different noise settings and warm start ratios in $\{23.0, 46.0, 92.0\}$. All CB algorithms use ϵ-greedy with $\epsilon = 0.1$. In each of the above plots, the x axis represents scores, while the y axis represents the CDF values.

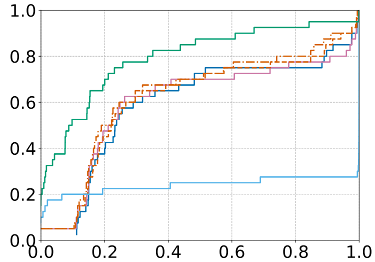
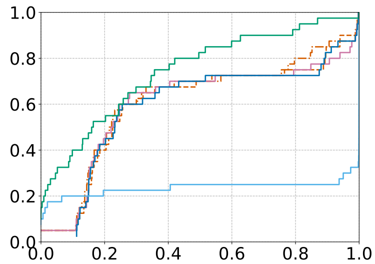
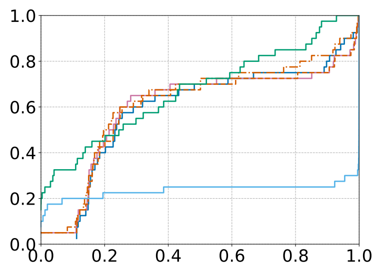
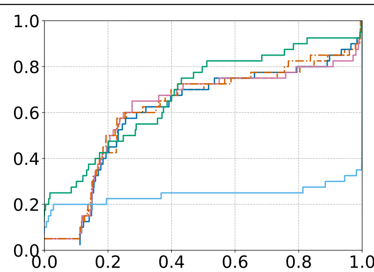
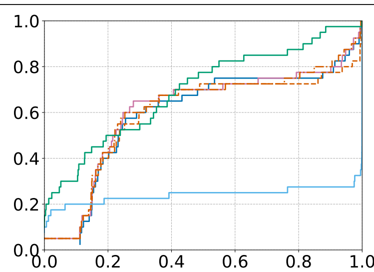
Noise	Ratio 184.0
Noiseless	
UAR $p = 0.25$	
MAJ $p = 0.25$	
CYC $p = 0.25$	
UAR $p = 0.5$	

Figure 18: Cumulative distribution functions (CDFs) for all evaluated algorithms, against different noise settings and warm start ratios in $\{184.0\}$. All CB algorithms use ϵ-greedy with $\epsilon = 0.1$. In each of the above plots, the x axis represents scores, while the y axis represents the CDF values.

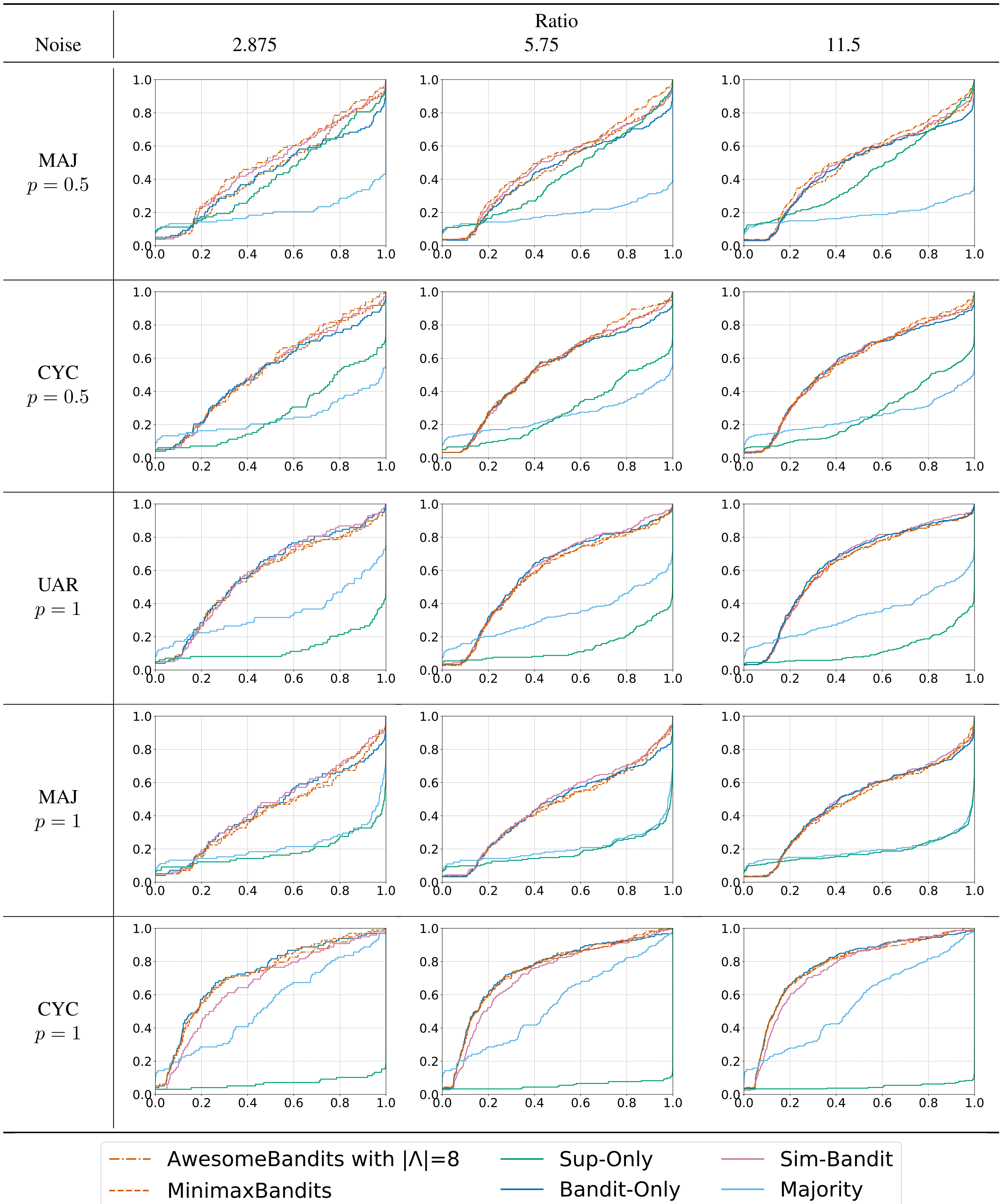


Figure 19: Cumulative distribution functions (CDFs) for all evaluated algorithms, against different noise settings and warm start ratios in $\{2.875, 5.75, 11.5\}$. All CB algorithms use ϵ-greedy with $\epsilon = 0.1$. In each of the above plots, the x axis represents scores, while the y axis represents the CDF values.

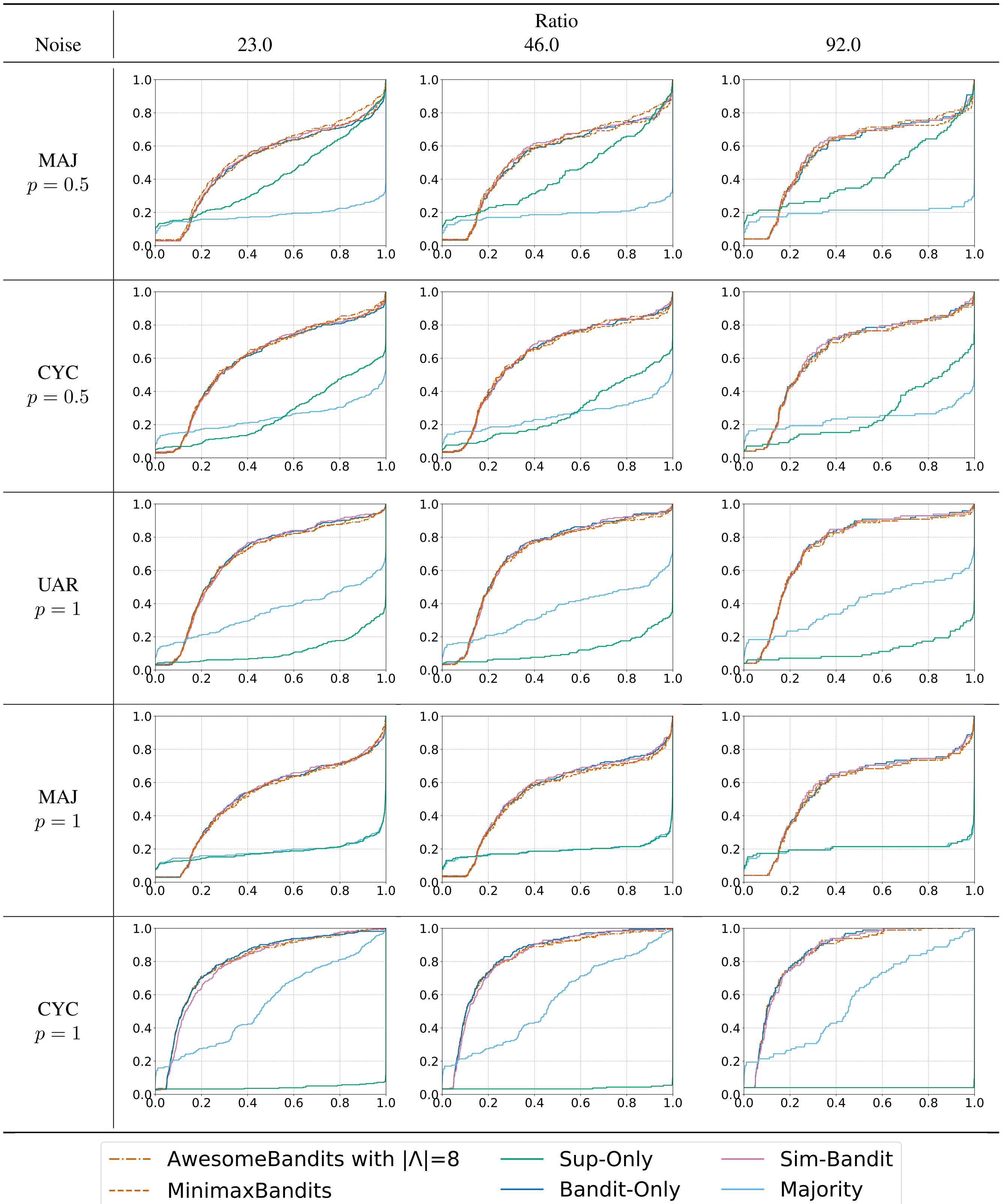
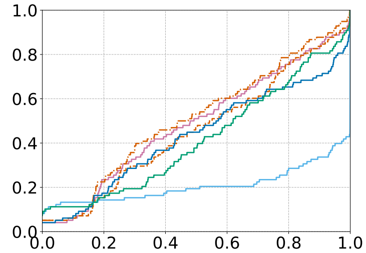
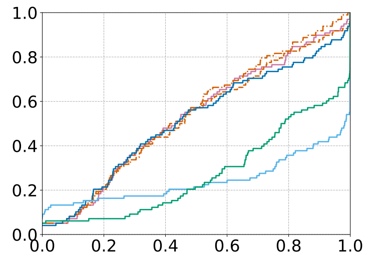
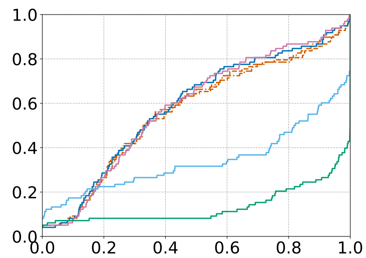
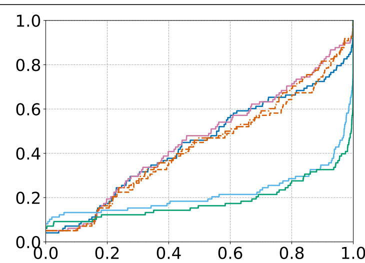
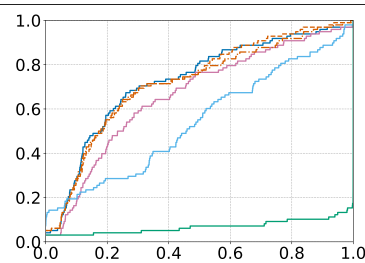


Figure 20: Cumulative distribution functions (CDFs) for all evaluated algorithms, against different noise settings and warm start ratios in $\{23.0, 46.0, 92.0\}$. All CB algorithms use ϵ-greedy with $\epsilon = 0.1$. In each of the above plots, the x axis represents scores, while the y axis represents the CDF values.

Noise	Ratio 184.0
MAJ $p = 0.5$	
CYC $p = 0.5$	
UAR $p = 1$	
MAJ $p = 1$	
CYC $p = 1$	

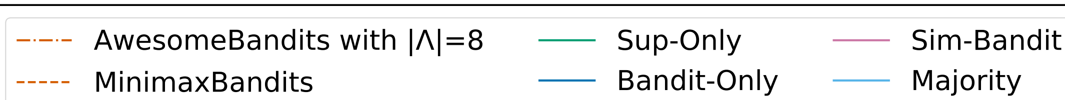


Figure 21: Cumulative distribution functions (CDFs) for all evaluated algorithms, against different noise settings and warm start ratios in $\{184.0\}$. All CB algorithms use ϵ-greedy with $\epsilon = 0.1$. In each of the above plots, the x axis represents scores, while the y axis represents the CDF values.