Data-Driven Online Model Selection With Regret Guarantees

Aldo Pacchiano
Boston University
Broad Institute of MIT and Harvard

Christoph Dann
Google Research

Claudio Gentile
Google Research

Abstract

We consider model selection for sequential decision making in stochastic environments with bandit feedback, where a meta-learner has at its disposal a pool of base learners, and decides on the fly which action to take based on the policies recommended by each base learner. Model selection is performed by regret balancing but, unlike the recent literature on this subject, we do not assume any prior knowledge about the base learners like candidate regret guarantees; instead, we uncover these quantities in a data-driven manner. The meta-learner is therefore able to leverage the *realized* regret incurred by each base learner for the learning environment at hand (as opposed to the *expected* regret), and single out the best such regret. We design two model selection algorithms operating with this more ambitious notion of regret and, besides proving model selection guarantees via regret balancing, we experimentally demonstrate the compelling practical benefits of dealing with actual regrets instead of candidate regret bounds.

1 INTRODUCTION

In online model selection for sequential decision making, the learner has access to a set of base learners and the goal is to adapt during learning to the best base learner that is the most suitable for the current environment. The set of base learners typically comes from instantiating different modelling assumptions or hyper-parameter choices, e.g., complexity of the reward model or the ϵ-parameter in ϵ-greedy. Which choice, and therefore which base learner, works best is highly dependent on the problem instance at hand, so that good online model selection solutions are important for robust sequential decision making. This has motivated an extensive study of model selection questions (e.g., Agarwal et al., 2017; Abbasi-Yadkori et al., 2020; Ghosh et al., 2020; Chatterji et al., 2020; Bibaut et al., 2020; Foster et al., 2020; Lee et al., 2020; Wei et al., 2022, and others cited below) in bandit and reinforcement learning problems. While some of these works have developed custom solutions for specific model selection settings, for instance, selecting among a nested set of linear policy classes in contextual bandits (e.g. Foster et al., 2019), the relevant literature also provides several general purpose approaches that work in a wide range of settings. Among the most prominent ones are FTRL-based (follow-the-regularized-leader) algorithms, including EXP4 (Odalric and Munos, 2011), Corral (Agarwal et al., 2017; Pacchiano et al., 2020b) and Tsallis-INF (Arora et al., 2020), as well as algorithms based on regret balancing (Abbasi-Yadkori et al., 2020; Pacchiano et al., 2020a; Cutkosky et al., 2021; Pacchiano et al., 2022).

These methods usually come with theoretical guarantees of the following form: the *expected regret* (or *high-probability regret*) of the model selection algorithm is not much worse than the expected regret (or high probability regret) of the best base learner. Such results are reasonable and known to be unimprovable in the worst-case (Marinov and Zimmert, 2021). Yet, it is possible for model selection to achieve expected regret that is systematically smaller than that of any base learner. This may seem surprising at first, but it can be explained through an example when considering the large variability across individual runs of each base learner on the same environment.

The situation is illustrated in Figure 1. On the left, we plot the cumulative expected regret of two base learners, along with the corresponding behavior of one of our model selection algorithms (ED^{2}RB – see Section 3.2 below) run on top of them. On the right, we unpack the cumulative expected regret curve of one of

Proceedings of the 27$^{\text{th}}$ International Conference on Artificial Intelligence and Statistics (AISTATS) 2024, Valencia, Spain. PMLR: Volume 238.

the two base learners from the left plot, and display ten independent runs of this base learner on the same environment, together with the resulting expected regret curve (first 1000 rounds only). Since the model selection algorithm has access to two base learners simultaneously, it can leverage a good run of either of two, and thereby achieve a good run more likely than any base learner individually, leading to overall smaller expected regret.

Such high variability in performance across individual runs of a base learner is indeed fairly common in model selection, for instance when base learners correspond to different hyper-parameters that control the explore-exploit trade-off. For a hyper-parameter setting that explores too little for the given environment, the base learner becomes unreliable and either is lucky and converges quickly to the optimal solution or unlucky and gets stuck in a suboptimal one. This phenomenon is a key motivation for our work. Instead of model selection methods that merely compete with the expected regret of any base learner, we design model selection solutions that compete with the regret *realizations* of any base learner, and have (data-dependent) theoretical guarantees that validate this ability.

While the analysis of FTRL-based model selection algorithms naturally lends itself to work with expected regret (e.g. Agarwal et al., 2017), the existing guarantees for regret balancing work with realized regret of base learners (e.g. Pacchiano et al., 2020a; Cutkosky et al., 2021). Concretely, regret balancing requires each learner to be associated with a *candidate* regret bound, and the model selection algorithm competes with the regret bound of the best among the well-specified learner, those learners whose regret realization is below their candidate bound. Setting a-priori tight candidate regret bounds for base learners is a main limitation for existing regret balancing methods, as the resolution of these bounds is often the one provided by a (typically coarse) theoretical analysis. As suggested in earlier work, we can create several copies of each base learner with different candidate bounds, but we find this not to perform well in practice due to the high number of resulting base learners. Another point of criticism for existing regret balancing methods is that, up to deactivation of base learners, these methods do not adapt to observations, since their choice among active base learners is determined solely by the candidate regret bounds, which are set a-priori.

In this work, we address both limitations, and propose two new regret balancing algorithms for model selection with bandit feedback that do not require knowing candidate regret bounds. Instead, the algorithms determine the right regret bounds sequentially in a data-driven manner, allowing them to adapt to the regret realization of the best base learner. We prove this by deriving regret guarantees that share the same form with existing theoretial results, but replace expected regret rates or well-specified regret bounds with realized regret rates, which can be much sharper (as in the example in Figure 1). From an empirical standpoint, we illustrate the validity of our approach by carrying out an experimental comparison with competing approaches to model selection via base learner pooling, and find that our new algorithms systematically outperform the tested baselines.

2 SETUP AND NOTATION

We consider a general sequential decision making framework that covers many important problem classes such as multi-armed bandits, contextual bandits and tabular reinforcement learning as special cases. This framework or variations of it has been commonly used in model selection (e.g. Cutkosky et al., 2021; Wei et al., 2022; Pacchiano et al., 2022).

The learner operates with a policy class Π and a set of contexts $\mathcal{X}$ over which is defined a probability distribution $\mathcal{D}$, unknown to the learner. In bandit settings, each policy π is a mapping from contexts $\mathcal{X}$ to $\Delta_{\mathcal{A}}$, where $\mathcal{A}$ is an action space and $\Delta_{\mathcal{A}}$ denotes the set of probability distributions over $\mathcal{A}$. However, the concrete form of Π, $\mathcal{X}$ or $\mathcal{A}$ is not relevant for our purposes. We only need that each policy $\pi \in \Pi$ is associated with a fixed expected reward mapping $\mu^\pi \colon \mathcal{X} \to [0,1]$ of the form $\mu^\pi(x) = \mathbb{E}[r|x,\pi]$, which is unknown to the learner. In each round $t \in \mathbb{N}$ of the sequential decision process, the learner first decides on a policy $\pi_t \in \Pi$. The environment then draws a context $x_t \sim \mathcal{D}$ as well as a reward observation $r_t \in [0,1]$ such that $\mathbb{E}[r_t|x_t,\pi_t] = \mu^{\pi_t}(x_t)$. The learner receives (x_t, r_t) before the next round starts.

We call $v^\pi = \mathbb{E}_{x\sim\mathcal{D}}[\mu^\pi(x)]$ the *value* of a policy $\pi \in \Pi$ and define the instantaneous regret of π as

$$\mathrm{reg}(\pi) = v^\star - v^\pi = \mathbb{E}_{x\sim\mathcal{D}}[\mu^{\pi_\star}(x) - \mu^\pi(x)] \quad (1)$$

where $\pi_\star \in \mathrm{argmax}_{\pi\in\Pi}\, v^\pi$ is an optimal policy and $v^\star$ its value. The total regret after T rounds of an algorithm that chooses policies $\pi_1, \pi_2, \dots$ is $\mathrm{Reg}(T) = \sum_{t=1}^T \mathrm{reg}(\pi_t)$. Note that $\mathrm{Reg}(T)$ is a random quantity since the policies π_t selected by the algorithm depend on past observations, which are themselves random variables. Yet, we use in (1) a pseudo-regret notion that takes expectation over reward realizations and context draws. This is most convenient for our purposes but we can achieve guarantees without those expectations by paying an additive $O(\sqrt{T})$ term, as is standard. We also use $u_T = \sum_{t=1}^T v^{\pi_t}$ for the total value accumulated by the algorithm over the T rounds.

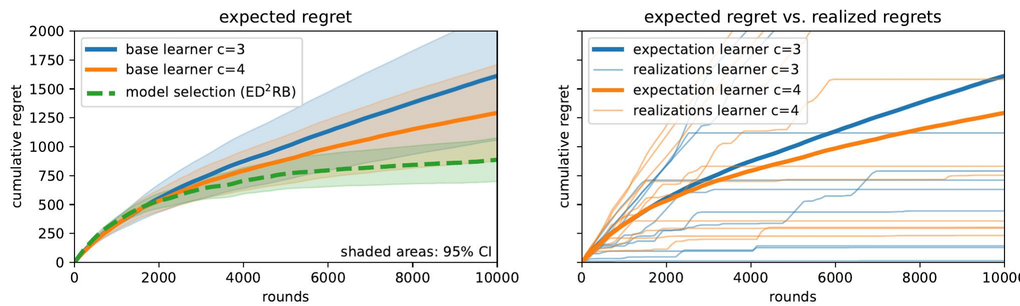


Figure 1: **Left:** Expected regret of two base learners (UCB on MAB with confidence scaling c controlling explore-exploit trade-off) and a model selection algorithm on top of them. The model selection algorithm has smaller expected regret than any base learner. **Right:** Expected regret and individual regret realizations (independent sample runs) of base learners. The base learners have highly variable performance which model selection can capitalize on. Detailed setup in Appendix 7.

Base learners. The learner (henceforth called *meta-learner*) is in turn given access to M base learners that the meta-learner can consult when determining the current policy to deploy. Specifically, in each round t, the meta-learner chooses one base learner $i_t \in [M] = \{1, \dots, M\}$ to follow and plays the policy suggested by this base learner. The policy that base learner i recommends in round t is denoted by π_t^i and thus $\pi_t = \pi_t^{i_t}$. We shall assume that each base learner has an internal state (and internal clock) that gets updated only on the rounds where that base learner is chosen. After being selected in round t, base learner i_t will receive from the meta-learner the observation (x_t, r_t). We use $n_t^i = \sum_{\ell=1}^t \mathbf{1}\{i_t = i\}$ to denote the number of times base learner i happens to be chosen up to round t, and by $u_t^i = \sum_{\ell=1}^t \mathbf{1}\{i_t = i\} v^{\pi_t^i}$ the total value accumulated by base learner i up to this point. It is sometimes more convenient to use a base learner's internal clock instead of the total round index t. To do so, we will use subscripts (k) with parentheses to denote the internal time index of a specific base learner, while subscripts t refer to global round indices. For example, given the sequence of realizatons $(x_1, r_1), (x_2, r_2), \dots,$ $\pi_{(k)}^i$ is the policy base learner i wants to play when being chosen the k-th time, i.e., $\pi_t^i = \pi_{(n_t^i)}^i$. The total regret incurred by a meta-learner that picks base learners $i_1, \dots, i_T$ can then be decomposed into the sum of regrets incurred by each base learner:

$$\mathrm{Reg}(T) = \sum_{t=1}^{T} \mathrm{reg}(\pi_t) = \sum_{i=1}^{M} \sum_{k=1}^{n_T^i} \mathrm{reg}(\pi_{(k)}^i).$$

2.1 Data-Driven Model Selection

Our goal is to perform model selection in this setting: We devise sequential decision making algorithms that have access to base learners as subroutines and are guaranteed to have regret that is comparable to the smallest *realized* regret, among all base learners in the pool, despite not knowing a-priori which base learner will happen to be best for the environment at hand ($\mathcal{D}$ and μ^π), and the actual realizations $(x_1, r_1), (x_2, r_2), \dots, (x_T, r_T)$.

In order to better quantify this notion of realized regret, the following definition will come handy.

Definition 2.1 (regret scale and coefficients)**.** *The* regret scale *of base learner i after being played k rounds is $\frac{\sum_{\ell=1}^k \mathrm{reg}(\pi_{(\ell)}^i)}{\sqrt{k}}$. For a positive constant $d_{\min}$, the* regret coefficient *of base learner i after being played k rounds is defined as*

$$d_{(k)}^i = \max\left\{\frac{\sum_{\ell=1}^k \mathrm{reg}(\pi_{(\ell)}^i)}{\sqrt{k}}, d_{\min}\right\}.$$

That is, $d_{(k)}^i \geq d_{\min}$ is the smallest number such that the incurred regret is bounded as $\sum_{\ell=1}^k \mathrm{reg}(\pi_{(\ell)}^i) \leq d_{(k)}^i \sqrt{k}$. Further we define the monotonic regret coefficient *of base learner i after being played k rounds as $\bar{d}_{(k)}^i = \max_{\ell \in [k]} d_{(\ell)}^i$.*

We use a $\sqrt{k}$ rate in this definition since that is the most commonly targeted regret rate in stochastic settings. Our results can be adapted, similarly to prior work (Pacchiano et al., 2020a) to other rates but the $\sqrt{T}$ barrier for model selection (Pacchiano et al., 2020b) remains of course.

It is worth emphasizing that both $d_{(k)}^i$ and $\bar{d}_{(k)}^i$ in the Definition 2.1 are random variables depending on $(x_1, r_1), (x_2, r_2), \dots, (x_\ell, r_\ell)$, where $\ell = \min\{t : n_t^i = k\}$. We illustrate them in Figure 2.

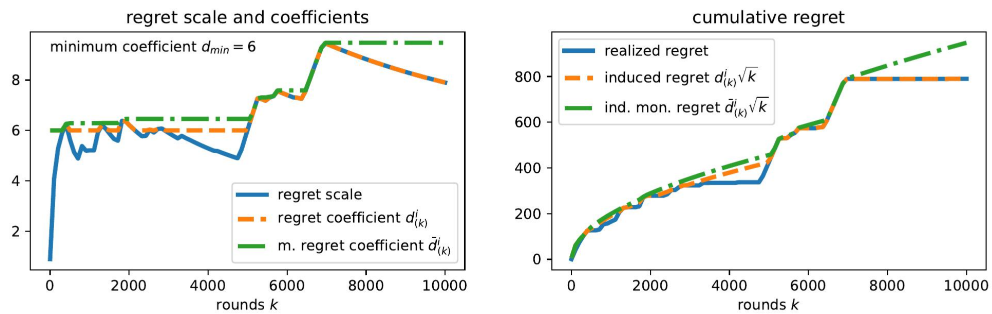


Figure 2: Illustration of Definition 2.1 for one of the baseline realizations from Figure 1. **Left:** Evolution of regret scale, coefficient and monotonic coefficient. **Right:** The same curves multiplied by $\sqrt{k}$. The induced regret bounds from regret coefficients follow the realized regret closely, the non-monotonic version more closely than the monotonic.

2.2 Running Examples

The above formalization encompasses a number of well-known online learning frameworks, including finite horizon Markov decision processes and contextual bandits, and model selection questions therein. We now introduce two examples but refer to earlier works on model selection for a more exhaustive list (e.g. Cutkosky et al., 2021; Wei et al., 2022; Pacchiano et al., 2022).

Tuning UCB exploration coefficient in multi-armed-bandits. As a simple illustrative example, we consider multi-armed bandits where the learner chooses in each round an action a_t from a finite action set $\mathcal{A}$ and receives a reward r_t drawn from a distribution with mean μ^{a_t} and unknown but bounded variance σ^2. In this setting, we directly identify each policy with an action, i.e., $\Pi = \mathcal{A}$ and define the context $\mathcal{X} = \{\varnothing\}$ as empty. The value of an action / policy a is simply $v^a = \mu^a$.

The variance σ strongly affects the amount of exploration necessary, thereby controlling the difficulty or "complexity" of the learning task. Since the explore-exploit of a learner is typically controlled through a hyper-parameter, it is beneficial to perform model selection among base learners with different trade-offs to adapt to the right complexity of the environment at hand. We use a simple UCB strategy as a base learner that chooses the next action as $\operatorname{argmax}_{a\in\mathcal{A}} \hat{\mu}(a) + c\sqrt{\frac{\ln(n(a)/\delta)}{n(a)}}$ where $n(a)$ and $\hat{\mu}(a)$ are the number of pulls of arm a so far and the average reward observed. Here c is the confidence scaling and we instantiate different base learners $i \in [M]$ with different choices $c_1, \dots, c_M$ for c. The goal is to adapt to the best confidence scaling $c_{i_\star}$, without knowing the true variance σ^2.[1]

Nested linear bandits. In the stochastic linear bandit model, the learner chooses an action $a_t \in \mathcal{A}$ from a large but finite action set $\mathcal{A} \subset \mathbb{R}^d$, for some dimension $d > 0$ and receives as reward $r_t = a_t^\top \omega$ + white noise, where $\omega \in \mathbb{R}^d$ is a fixed but unknown reward vector. This fits in our framework by considering policies of the form $\pi_\theta(x) = \operatorname{argmax}_{a\in\mathcal{A}}\langle a, \theta\rangle$ for a parameter $\theta \in \mathbb{R}^d$, defining contexts $\mathcal{X} = \{\varnothing\}$ as empty and the mean reward as $\mu^\pi(x) = \pi(x)^\top\omega$, which is also the value v^π.

We here consider the following model selection problem, that was also a motivating application in Cutkosky et al. (2021). The action set $\mathcal{A} \subset \mathbb{R}^{d_M}$ has some maximal dimension $d_M > 0$, and we have an increasing sequence of M dimensions $d^1 < \ldots < d^M$. Associated with each d^i is a base learner that only considers policies Π_i of the form $\pi_{\theta_i}(x) = \operatorname{argmax}_{a\in\mathcal{A}}\langle P_{d_i}[a], \theta_i\rangle$ for $\theta_i \in \mathbb{R}^{d^i}$ and $P_{d^i}[\cdot]$ being the projection onto the first d^i dimensions. That is, the i-th base learner operates only on the first d^i components of the unknown reward vector $\omega \in \mathbb{R}^{d^M}$. If we stipulate that only the first $d^{i_\star}$ dimensions of $\omega \in \mathbb{R}^{d^M}$ are non-zero ($d^{i_\star}$ being unknown to the learner) we are in fact competing in a regret sense against the base learner that operates with the policy class $\Pi_{i_\star}$, the one at the "right" level of complexity for the underlying ω.

Nested stochastic linear contextual bandits. We also consider a contextual version of the previous setting (Lattimore and Szepesvári, 2020, Ch. 19) where where context $x_t \in \mathcal{X}$ are drawn i.i.d. and which a policy maps to some action $a_t \in \mathcal{A}$. The expected reward is then $\mu^\pi(x) = \psi(x, \pi(x))^\top\omega$ for a known fea-

[1] We choose this example for its simplicity. An alternative without model selection would be UCB with empirical Bernstein confidence bounds (Audibert et al., 2007). However, adaptation with model selection works just as well in more complex settings e.g. linear bandits and MDP, where empirical variance confidence bounds are not available or much more complicated.

ture embedding $\psi : \mathcal{X} \times \mathcal{A} \to \mathcal{R}^d$, and an unknown vector $\omega \in \mathcal{R}^d$. Just as above, we consider the nested version of this setting where ψ and ω live in a large ambient dimension d^M but only the first $d^{i_\star}$ entries of ω are non-zero.

3 DATA-DRIVEN REGRET BALANCING

We introduce and analyze two data-driven regret balancing algorithms, which are both shown in Algorithm 1. Both algorithms maintain over time three main estimators for each base learner: (1) regret coefficients $\widehat{d}_t^i$, meant to estimate the monotonic regret coefficients $\bar{d}_t^i$ from Definition 2.1, (2) the average reward estimators $\widehat{u}_t^i/n_t^i$, and (3) the balancing potentials ϕ_t^i, which are instrumental in the implementation of the exploration strategy based on regret balancing. At each round t the meta-algorithm picks the base learner i_t with the smallest balancing potential so far (ties broken arbitrarily). The algorithm plays the policy π_t suggested by that base learner on the current context x_t, receives the associated reward r_t, and forwards (x_t, r_t) back to that base learner only.

Where our two meta-learners differ is how they update the regret coefficient $\widehat{d}_t^{i_t}$ of the chosen learner and its potential $\phi_t^{i_t}$. We now introduce each version and the regret guarantee we prove for it.

3.1 Balancing Through Doubling

Our first meta-algorithm (Doubling Data Driven Regret Balancing or D^{3}RB) is shown on the left in Algorithm 1. Similar to existing regret balancing approaches (Pacchiano et al., 2020b, 2022), D^{3}RB performs a misspecification test which checks whether the current estimate of the regret of base learner i_t is compatible with the data collected so far. The test compared the average reward $\frac{\widehat{u}_t^{i_t}}{n_t^{i_t}}$ of the chosen learner against the highest average reward among all learners $\max_{j\in[M]} \frac{\widehat{u}_t^j}{n_t^j}$. If the difference is larger than the current regret coefficient $\frac{\widehat{d}_{t-1}^{i_t}\sqrt{n_t^{i_t}}}{n_t^{i_t}}$ permits (accounting for estimation errors by considering appropriate concentration terms), then the we know that $\widehat{d}^{i_t}$ is too small to accurately represent the regret of learner i_t and we double it. This deviates from prior regret balancing approaches (Pacchiano et al., 2020b; Cutkosky et al., 2021) that simply eliminate a base learner if the misspecification test fails for a given candidate regret bound. Finally, D^{3}RB sets the potential $\phi_t^{i_t}$ as $\widehat{d}_t^{i_t}\sqrt{n_t^{i_t}}$ so that the potential represents an upper-bound on the regret incurred by i_t.

Our doubling approach for $\widehat{d}_t^{i_t}$ is algorithmically simple but creates main technical hurdles compared to existing elimination approaches since we have to show that the regret coefficients are adapted fast enough to be accurate and do not introduce undesirable scalings in our upper bounds. By overcoming these hurdles in our analysis, we show the following result quantifies the regret properties of D^{3}RB in terms of the *monotonic* regret coefficients of the base learners at hand.

Theorem 3.1. *With probability at least $1-\delta$, the regret of D^{3}RB (Algorithm 1, left) with parameters δ and $d_{\min} \geq 1$ is bounded in all rounds $T \in \mathbb{N}$ as*[2]

$$\mathrm{Reg}(T) = \tilde{O}\left(\bar{d}_T^\star M\sqrt{T} + (\bar{d}_T^\star)^2\sqrt{MT}\right)$$

where $\bar{d}_T^\star = \min_{i\in[M]} \bar{d}_T^i = \min_{i\in[M]} \max_{t\in[T]} d_t^i$ is the smallest monotonic regret coefficient among all learners (see Definition 2.1).

We defer a discussion of this regret bound and comparison to existing results to Section 3.3.

3.2 Balancing Through Estimation

While D^{3}RB retains the misspecification test of existing regret balancing approaches, our second algorithm, Estimating Data-Driven Regret Balancing or ED^{2}RB, takes a more direct approach. It estimates the regret coefficient (see right in Algorithm 1) directly as the highest difference in average reward $\max_{j\in[M]} \frac{\widehat{u}_t^j}{n_t^j} - \frac{\widehat{u}_t^{i_t}}{n_t^{i_t}}$ between i_t and any other learner scaled by $\sqrt{n_t^{i_t}}$, the number of times i_t has been played. Again, we include appropriate concentration terms to account for estimation errors as well as a lower bound of $d_{\min}$ to ensure stability.

Since this direct estimation approach is more adaptive compared to the doubling approach, it requires a finer analysis to show that the changes in the estimator does not interfere with the balancing property of the algorithm. To ensure the necessary stability, we use a clipped version in ED^{2}RB of the potential of D^{3}RB. The function $\mathrm{clip}(x; a, b)$ therein clips the real argument x to the interval $[a, b]$, and makes the potential non-decreasing and not increasing too quickly.

This more careful definition of the balancing potentials allows us to replace in the regret bound the monotonic regret coefficient $\bar{d}_T^\star$ with the sharper regret coefficient $d_T^\star$ in the regret guarantee for ED^{2}RB:

Theorem 3.2. *With probability at least $1-\delta$, the regret of ED^{2}RB (Algorithm 1, right) with parameters δ and $d_{\min} \geq 1$ is bounded in all rounds $T \in \mathbb{N}$ as*

$$\mathrm{Reg}(T) = \tilde{O}\left(d_T^\star M\sqrt{T} + (d_T^\star)^2\sqrt{MT}\right)$$

[2] Here and throughout, $\tilde{O}$ hides log-factors.

Algorithm 1: Data Driven Regret Balancing Algorithms (D^{3}RB and ED^{2}RB)

Input: M base learners, minimum regret coefficient $d_{\min}$, failure probability δ
Initialize balancing potentials $\phi_1^i = d_{\min}$ and regret coefficient $\widehat{d}_0^i = d_{\min}$, for all $i \in [M]$
Initialize counts $n_0^i = 0$, and total value $\widehat{u}_0^i = 0$, for all $i \in [M]$
for *rounds* $t = 1, 2, 3, \ldots$ **do**
    Receive context x_t
    Pick base learner i_t with smallest balancing potential: $i_t \in \operatorname{argmin}_{i\in[M]} \phi_t^i$
    Pass x_t to base learner i_t
    Play policy $\pi_t = \pi_t^{i_t}$ suggested by base learner i_t on x_t and receive reward r_t
    Set $n_t^i = n_{t-1}^i$, $\widehat{u}_t^i = \widehat{u}_{t-1}^i$, $\widehat{d}_t^i = \widehat{d}_{t-1}^i$, and $\phi_{t+1}^i = \phi_t^i$, for $i \in [M] \setminus \{i_t\}$
    Update statistics $n_t^{i_t} = n_{t-1}^{i_t} + 1$ and $\widehat{u}_t^{i_t} = \widehat{u}_{t-1}^{i_t} + r_t$

D^{3}RB	or	ED^{2}RB
Perform misspecification test $$\frac{\widehat{u}_t^{i_t}}{n_t^{i_t}} + \frac{\widehat{d}_{t-1}^{i_t}\sqrt{n_t^{i_t}}}{n_t^{i_t}} + c\sqrt{\frac{\ln\frac{M\ln n_t^{i_t}}{\delta}}{n_t^{i_t}}} < \max_{j\in[M]} \frac{\widehat{u}_t^j}{n_t^j} - c\sqrt{\frac{\ln\frac{M\ln n_t^j}{\delta}}{n_t^j}}$$ If test triggered double regret coefficient $\widehat{d}_t^{i_t} = 2\widehat{d}_{t-1}^{i_t}$ and otherwise set $\widehat{d}_t^{i_t} = \widehat{d}_{t-1}^{i_t}$ Update balancing potential $\phi_{t+1}^{i_t} = \widehat{d}_t^{i_t}\sqrt{n_t^{i_t}}$		Estimate active regret coefficient $$\widehat{d}_t^{i_t} = \max\left\{d_{\min},\ \sqrt{n_t^{i_t}}\left(\max_{j\in[M]} \frac{\widehat{u}_t^j}{n_t^j} - c\sqrt{\frac{\ln\frac{M\ln n_t^j}{\delta}}{n_t^j}} - \frac{\widehat{u}_t^{i_t}}{n_t^{i_t}} - c\sqrt{\frac{\ln\frac{M\ln n_t^{i_t}}{\delta}}{n_t^{i_t}}}\right)\right\}$$ Update balancing potential $$\phi_{t+1}^{i_t} = \operatorname{clip}\left(\widehat{d}_t^{i_t}\sqrt{n_t^{i_t}};\ \phi_t^{i_t},\ 2\phi_t^{i_t}\right)$$

where $d_T^\star = \min_{i\in[M]} \max_{j\in[M]} d_{T_j}^i$ is the smallest regret coefficient among all learners, and T_j is the last time t when base learner j was played and $\phi_{t+1}^j < 2\phi_t^j$.

3.3 Discussion, Comparison to the Literature

One way to interpret Theorem 3.1 is the following. If the meta-learner were given ahead of time the index of the base learner achieving the smallest monotonic regret coefficient $\bar{d}_T^\star$, then the meta-learner would follow that base learner from beginning to end. The resulting regret bound for the meta-learner would be of the form $(\bar{d}_T^\star)\sqrt{T}$. Then the price D^{3}RB pays for aggregating the M base learners is essentially a multiplicative factor of the form $M + \bar{d}_T^\star\sqrt{M}$.

Up to the difference between $d_T^\star$ and $\bar{d}_T^\star$, the guarantees in Theorem 3.1 and Theorem 3.2 are identical. Further, since $d_T^\star \le \bar{d}_T^\star$, the guarantee for ED^{2}RB is never worse than that for D^{3}RB. It can however be sharper, e.g., in environments with favorable gaps where we expect that a good base learner may achieve a $O(\log(T))$ regret instead of a $\sqrt{T}$ rate and thus d_t^i of that learner would decrease with time. The regret coefficient $d_T^\star$ can benefit from this while $\bar{d}_T^\star$ cannot decrease with T, and thus provide a worse guarantee.

Both D^{3}RB and ED^{2}RB rely on a user-specified parameter $d_{\min}$. In terms of regret coefficients, the regret bounds of the two algorithms have the general form $((d_T^\star)^2/d_{\min} + d_{\min})\sqrt{T}$. To see this, one has to observe that, the way it is defined, the regret coefficient $d_{(k)}^i$ satisfies $d_{(k)}^i = \Theta\Big(\frac{\sum_{\ell=1}^k \operatorname{reg}(\pi_{(\ell)}^i)}{\sqrt{k}} + d_{\min}\Big)$, and then take a closer look at the proof of Theorem 3.2, just before the line "Plugging in $d_{min} \ge 1$ gives" therein. A similar observation applies to the monotone version $\bar{d}_{(k)}^i$ and the proof of Theorem 3.1. So, if we knew beforehand something about $d_T^\star$, we could set $d_{\min} = d_T^\star$, and get a linear dependence on $d_T^\star$, otherwise we can always set as default $d_{\min} = 1$ (as we did in Theorem 3.1 and Theorem 3.2).

Both our data-dependent guarantees recover existing data-*independent* results up to the precise M dependency. Specifically, ignoring M factors, our bounds scale at most as $(d_T^\star)^2\sqrt{T}$, while the previous literature on the subject (e.g., Cutkosky et al. (2021), Corollary 2) scales as $(d^{i_\star})^2\sqrt{T}$. We recall that in the data-independent case, regret *lower* bounds are contained (in a somewhat implicit form) in Marinov and Zimmert (2021), where it is shown that one cannot hope in general to achieve better results in terms of $d^{i_\star}$ and T, like in particular a $d^{i_\star}\sqrt{T}$ regret bound. Only a $(d^{i_\star})^2\sqrt{T}$-like regret guarantee is generally possible.

These prior model selection results based on regret balancing require candidate regret bounds to be specified ahead of time. Hence, the corresponding algorithms cannot leverage the favorable cases that our data-dependent bounds automatically adapt to. In particular, in Cutkosky et al. (2021), the optimal parameter $d^{i_\star}$ is the smallest candidate regret rate that is larger than the true rate of the optimal (well-specified) base learner. Instead, we do not assume the availability of such candidate regret rates, and our $d_T^\star$ is the *true* regret rate of the optimal base learner. In short, Cutkosky et al. (2021)'s results are competitive with ours only when the above candidate regret rate happens to be very accurate for the best base learner, but this is a fairly strong assumption. Although theoretical regret bounds for base learners can often guide the guess for the regret rate, the values one obtains from those analyses are typically much larger than the true regret rate, as theoretical regret bounds are usually loose by large constants.

So, the goal here is to improve over more traditional data-independent bounds when data is benign or typical. Observe that in practice $d_T^\star$ can also be *decreasing* with T (as we will show multiple times in our experiments in Section 4). One such relevant case is when the individual base learner runs have large variances (recall Figure 1).

From a technical standpoint, we do indeed build on the existing technique for analyzing regret balancing by Pacchiano et al. (2020b); Cutkosky et al. (2021). Yet, their analysis heavily relies on fixed candidate regret bounds, and removing those introduces several technical challenges, like disentangling the balancing potentials ϕ_t^i from the estimated regret coefficients $\widehat{d}_t^i$, and combine with clipping or the doubling estimator. This allows us to show the necessary invariance properties that unlocks our improved data-dependent guarantees. See Appendix 9 and 10.

Departing from regret balancing techniques, model selection can also revolve around Follow-The-Regularized Leader-like schemes (e.g., (Agarwal et al., 2017; Pacchiano et al., 2020b; Arora et al., 2020)). However, even in those papers, $d^{i_\star}$ is the expected regret scale, thus never sharper than our $d_T^\star$, and also not able to capture favorable realizations. As we shall see in Section 4, there is often a stark difference between the expected performance and the data-dependent performance, which confirms that the improvement in our bounds is important in practice.

4 EXPERIMENTS

We evaluate our algorithms on several synthetic benchmarks (environments, base-learners and model selection tasks), and compare their performance against existing meta-learners. For all details of the experimental setup and additional results, see Appendix 11.

These experiments are mostly intended to validate the theory and as a companion to our theoretical results. In these experiments, we vary the parameters that we expect to be most important for model selection Varying the difficulty of the learning environment itself is something that should mostly be absorbed by base-learners, for example, by choosing base learners operating on more powerful function classes than we do here. Yet, it is important to observe that the meta-algorithms are fairly oblivious to the difficulty of the environment. All that matters here is the regret profile of the base learners. In our experiments, we therefore decided to explore the landscape of model selection by varying the nature of the model selection task itself (dimension, self model selection, and confidence scaling) while keeping the underlying environments fairly simple.

Environments and base-learners: As the first environment, we use a simple 5-armed multi-armed bandit problem (**MAB**) with standard Gaussian noise. We then use two linear bandit settings, as also described in Section 2.2: linear bandits with stochastic rewards, either with a stochastic context (**CLB**) or without (**LB**). As base learners, we use **UCB** for the MAB environment (see also Section 2.2) and Linear Thompson (**LinTS**) sampling (Abeille and Lazaric, 2017) for the LB and CLB setting.

Model selection task: We consider 3 different model selection tasks. All the results are reported in Figure 3. In the first, **conf** ("confidence"), we vary the explore-exploit trade-off in the base learners. For UCB, different base learners correspond to different settings of c, the confidence scaling that multiplies the exploration bonus. Analogously, for **LinTS**, we vary the scale c of the parameter perturbation. For the second task **dim** ("dimension"), we vary the number of dimensions d_i the base learner considers when choosing the action (see second and third example in Section 2.2, as well as Figure 3 for results). Finally, we also consider a "**self**" task, where all base learners are copies of the same algorithm.

Meta-learners: We evaluate both our algorithms, **D^{3}RB** and **ED^{2}RB**, from Algorithm 1. We com-

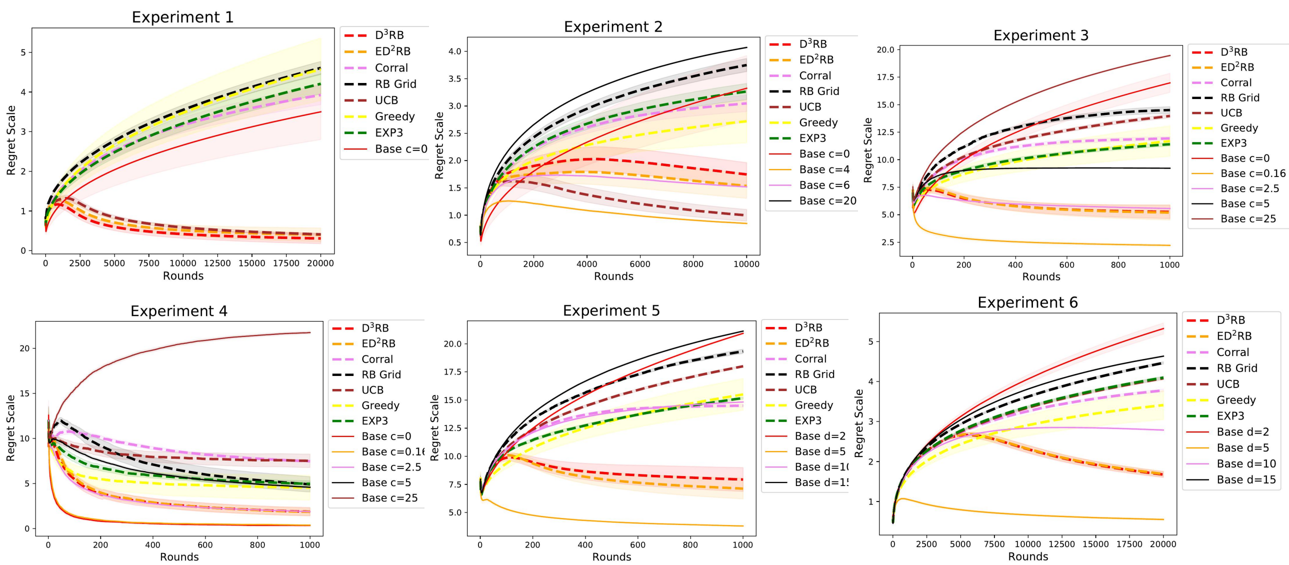


Figure 3: Average performance comparing all meta-learners (see Table 1 for reference). **Experiment 1:**. Self model selection. See also Figure 5 in Appendix 11.3, containing regret curves for **D^{3}RB** and **ED^{2}RB** on a single realization. **Experiment 2:** base learners (**UCB**) with different confidence multipliers c. **Experiments 3 and 4:** Dimensionality $d = 10$. **Experiments 5 and 6:** True dimensionality $d^{i_\star} = 5$ and maximal dimensionality $d_M = 15$. In **Experiments** 3 and 5 the action set is the unit sphere. In **Experiments** 4 and 6 the contexts x_t are 10 actions sampled uniformly from the unit sphere.

Table 1: Comparison of meta learners: cumulative regret (averaged over 100 repetitions $\pm 2\times$standard error) at the end of the sequence of rounds. In bold is the best performer for each environment.

	Env.	Learners	Task	D^{3}RB	ED^{2}RB	Corral	RB Grid	UCB	Greedy	EXP3
1.	MAB	UCB	self	$\mathbf{431 \pm 182}$	560 ± 240	5498 ± 340	6452 ± 230	574 ± 34	6404 ± 1102	5892 ± 356
2.	MAB	UCB	conf	1608 ± 198	1413 ± 208	2807 ± 138	3452 ± 110	$\mathbf{918 \pm 98}$	2505 ± 362	3007 ± 136
3.	LB	LinTS	conf	1150 ± 134	$\mathbf{1135 \pm 148}$	2605 ± 38	3169 ± 66	3052 ± 36	2553 ± 302	2491 ± 36
4.	CLB	LinTS	conf	411 ± 100	$\mathbf{406 \pm 94}$	1632 ± 30	1073 ± 184	1644 ± 160	991 ± 298	1086 ± 70
5.	LB	LinTS	dim	1733 ± 230	$\mathbf{1556 \pm 198}$	3166 ± 26	4223 ± 40	3932 ± 16	3385 ± 306	3315 ± 20
6.	CLB	LinTS	dim	$\mathbf{2347 \pm 102}$	2365 ± 96	5294 ± 44	6258 ± 38	5718 ± 50	4778 ± 506	5742 ± 46

pare them against the **Corral** algorithm (Agarwal et al., 2017) with the stochastic wrapper from Pacchiano et al. (2020b), as a representative for FTRL-based meta-learners. We also evaluate Regret Balanncing from Pacchiano et al. (2020b); Cutkosky et al. (2021) with several copies of each base learner, each with a different candidate regret bound, selected on an exponential grid (**RB Grid**). We also include in our list of competitors three popular algorithms, the **Greedy** algorithm (always selecting the best base learner so far with no exploration), **UCB** (Auer et al., 2002a) and **EXP3** (Auer et al., 2002b). These are legitimate choices as meta-algorithms, but either they do not come with theoretical guarantees in the model selection setting (UCB, Greedy) or enjoy worse guarantees (Pacchiano et al., 2020b).

Discussion. An overview of our results can be found in Table 1, where we report the cumulative regret of each algorithm at the end of each experiment. Figure 3 contains the entire learning curves (as regret scale = cumulative regret normalized by $\sqrt{T}$). We observe that D^{3}RB and ED^{2}RB both outperform all other meta-learners on all but the second benchmark. UCB as a meta-learner performs surprisingly well in benchmarks on MABs but performs poorly on the others.

This can be explained by observing that the regret of the optimal algorithms in the MAB environments quickly converges to a flat line. Thus, from the perspective of a UCB meta-learner, they look like stationary/stateless reward arms. This is likely the reason why UCB works so well in Environment 2. This situation should be contrasted to the regret curves in linear environments (Environments 3–6 in Table 1), which exhibit more diverse non-stationary behaviors.

Overall, our methods feature the smallest or close to the smallest cumulative regret among meta-learners on all benchmarks.

Comparing D^{3}RB and ED^{2}RB, we observe overall very similar performance, suggesting that ED^{2}RB may be preferable due to its sharper theoretical guarantee. While the model selection tasks conf and dim are standard in the literature, we also included one experiment with the self task where we simply select among different instances of the same base learner. This task was motivated by our initial observation (see also Figure 1) that base learners have often a very high variability between runs and that model selection can capitalize on. Indeed, Figure 3 shows that our algorithms as well as UCB achieve much smaller overall regret than the base learner. This suggests that model selection can be an effective way to turn a notoriously unreliable algorithm like the base greedy base learner (UCB with $c = 0$ is Greedy) into a robust learner.

5 CONCLUSIONS

We proposed two new algorithms for model selection based on the regret balancing principle but without the need to specify candidate regret bounds a-priori. This calls for more sophisticated regret balancing mechanics that makes our methods data-driven and as an important benefit allows them to capitalize on variability in a base learner's performance. We demonstrate this empirically, showing that our methods perform well across several synthetic benchmarks, as well as theoretically. We prove that both our algorithms achieve regret that is not much worse than the realized regret of any base learner. This data-dependent guarantee recovers existing data-independent results but can be significantly tighter.

In this work, we focused on the fully stochastic setting, with contexts and rewards drawn i.i.d. We believe an extension of our results to arbitrary contexts is fairly easy by replacing the deterministic balancing with a randomized version, while retaining the same definition of regret coefficients d_t^i as in the paper. Yet, this comes at a price of a harder interpretation of those coefficients. Whereas in the fully stochastic setting the regret coefficient d_t^i of base learner i is only a function of that base learner, in the more general adaptive setting, d_t^i would depend on the observed contexts, and thus potentially on the actions chosen by other base learners. To retain a clear interpretation, we therefore chose to only cover the i.i.d. stochastic context case.

On the other hand, covering the fully adversarial setting is likely possible by building on top of (Pacchiano et al., 2022) but requires substantial innovation.

6 ACKNOWLEDGEMENTS

A.P. would like to thank the support of the Eric and Wendy Schmidt Center at the Broad Institute of MIT and Harvard. This work was supported in part by funding from the Eric and Wendy Schmidt Center at the Broad Institute of MIT and Harvard.

References

Y. Abbasi-Yadkori, A. Pacchiano, and M. Phan. Regret balancing for bandit and rl model selection. *arXiv preprint arXiv:2006.05491*, 2020.

M. Abeille and A. Lazaric. Linear thompson sampling revisited. In *Artificial Intelligence and Statistics*, pages 176–184. PMLR, 2017.

A. Agarwal, H. Luo, B. Neyshabur, and R. E. Schapire. Corralling a band of bandit algorithms. In *Conference on Learning Theory*, pages 12–38. PMLR, 2017.

R. Arora, T. V. Marinov, and M. Mohri. Corralling stochastic bandit algorithms. *arXiv preprint arXiv:2006.09255*, 2020.

J.-Y. Audibert, R. Munos, and C. Szepesvari. Tuning bandit algorithms in stochastic environments. In *Algorithmic Learning Theory: 18th International Conference, ALT 2007, Sendai, Japan, October 1-4, 2007. Proceedings 18*, pages 150–165. Springer, 2007.

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

P. Auer, N. Cesa-Bianchi, Y. Freund, and R. E. Schapire. The nonstochastic multiarmed bandit problem. *SIAM Journal on Computing*, 32(1):48–77, 2002b.

A. F. Bibaut, A. Chambaz, and M. J. van der Laan. Rate-adaptive model selection over a collection of black-box contextual bandit algorithms. *arXiv preprint arXiv:2006.03632*, 2020.

N. Chatterji, V. Muthukumar, and P. Bartlett. Osom: A simultaneously optimal algorithm for multi-armed and linear contextual bandits. In *International Conference on Artificial Intelligence and Statistics*, pages 1844–1854, 2020.

A. Cutkosky, C. Dann, A. Das, C. Gentile, A. Pacchiano, and M. Purohit. Dynamic balancing for model selection in bandits and rl. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 2276–2285. PMLR, 2021.

D. Foster, C. Gentile, M. Mohri, and J. Zimmert. Adapting to misspecification in contextual bandits.

In *Advances in Neural Information Processing Systems*, 2020.

D. J. Foster, A. Krishnamurthy, and H. Luo. Model selection for contextual bandits. In *Advances in Neural Information Processing Systems*, pages 14741–14752, 2019.

A. Ghosh, A. Sankararaman, and K. Ramchandran. Problem-complexity adaptive model selection for stochastic linear bandits. *arXiv preprint arXiv:2006.02612*, 2020.

S. R. Howard, A. Ramdas, J. Mc Auliffe, and J. Sekhon. Uniform, nonparametric, non-asymptotic confidence sequences. *arXiv preprint arXiv:1810.08240*, 2018.

T. Lattimore and C. Szepesvári. *Bandit algorithms*. Cambridge University Press, 2020.

J. N. Lee, A. Pacchiano, V. Muthukumar, W. Kong, and E. Brunskill. Online model selection for reinforcement learning with function approximation. *arXiv preprint arXiv:2011.09750*, 2020.

T. V. Marinov and J. Zimmert. The pareto frontier of model selection for general contextual bandits. *Advances in Neural Information Processing Systems*, 34:17956–17967, 2021.

M. Odalric and R. Munos. Adaptive bandits: Towards the best history-dependent strategy. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, pages 570–578. JMLR Workshop and Conference Proceedings, 2011.

A. Pacchiano, C. Dann, C. Gentile, and P. Bartlett. Regret bound balancing and elimination for model selection in bandits and rl. *arXiv preprint arXiv:2012.13045*, 2020a.

A. Pacchiano, M. Phan, Y. Abbasi-Yadkori, A. Rao, J. Zimmert, T. Lattimore, and C. Szepesvari. Model selection in contextual stochastic bandit problems. *arXiv preprint arXiv:2003.01704*, 2020b.

A. Pacchiano, C. Dann, and C. Gentile. Best of both worlds model selection. In *Advances in Neural Information Processing Systems*, volume 35, pages 1883–1895. Curran Associates, Inc., 2022.

C.-Y. Wei, C. Dann, and J. Zimmert. A model selection approach for corruption robust reinforcement learning. In *International Conference on Algorithmic Learning Theory*, pages 1043–1096. PMLR, 2022.

Checklist

1. For all models and algorithms presented, check if you include:
   (a) A clear description of the mathematical setting, assumptions, algorithm, and/or model. **Yes, see Sect. 2 and Sect. 3**
   (b) An analysis of the properties and complexity (time, space, sample size) of any algorithm. **Yes, see Sect. 3**
   (c) (Optional) Anonymized source code, with specification of all dependencies, including external libraries. **No**

2. For any theoretical claim, check if you include:
   (a) Statements of the full set of assumptions of all theoretical results. **Yes, see Sect. 2 and Sect. 3**
   (b) Complete proofs of all theoretical results. **Yes, see Sect. 8-10 in the appendix**
   (c) Clear explanations of any assumptions. **Yes, see Sect. 2 and Sect. 3**

3. For all figures and tables that present empirical results, check if you include:
   (a) The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL). **Yes, Sect. 11 in the appendix contains detailed explanations of all our experimental results and hence ways to reproduce them**
   (b) All the training details (e.g., data splits, hyperparameters, how they were chosen). **Yes, see Appendix 11**
   (c) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). **Yes, Appendix 11**
   (d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). [Yes/No/Not Applicable] **Yes, see Appendix 11**

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:
   (a) Citations of the creator If your work uses existing assets. **Not Applicable**
   (b) The license information of the assets, if applicable. **Not Applicable**
   (c) New assets either in the supplemental material or as a URL, if applicable. **Not Applicable**
   (d) Information about consent from data providers/curators. **Not Applicable**

(e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content. **Not Applicable**

5. If you used crowdsourcing or conducted research with human subjects, check if you include:

   (a) The full text of instructions given to participants and screenshots. **Not Applicable**

   (b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. **Not Applicable**

   (c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. **Not Applicable**

APPENDIX

The appendix contains the extra material that was omitted from the main body of the paper.

7 DETAILS ON FIGURE 1

We consider a 5-armed bandit problem with rewards drawn from a Gaussian distribution with standard deviation 6 and mean $\frac{10}{10}, \frac{6}{10}, \frac{5}{10}, \frac{2}{10}, \frac{1}{10}$ for each arm respectively. We use a simple UCB strategy as a base learner that chooses the next action as $\operatorname{argmax}_{a \in \mathcal{A}} \hat{\mu}(a) + c\sqrt{\frac{\ln(n(a)/\delta)}{n(a)}}$ where $n(a)$ and $\hat{\mu}(a)$ are the number of pulls of arm a so far and the average reward observed. The base learners use $\delta = \frac{1}{10}$ and $c = 3$ or $c = 4$ respectively.

8 ANALYSIS COMMON TO BOTH ALGORITHMS

Definition 8.1. *We define the event $\mathcal{E}$ in which we analyze both algorithms as the event in which for all rounds $t \in \mathbb{N}$ and base learners $i \in [M]$ the following inequalities hold*

$$-c\sqrt{n_t^i \ln \frac{M \ln n_t^i}{\delta}} \leq \widehat{u}_t^i - u_t^i \leq c\sqrt{n_t^i \ln \frac{M \ln n_t^i}{\delta}}$$

for the algorithm parameter $\delta \in (0,1)$ and a universal constant $c > 0$.

Lemma 8.2. *Event $\mathcal{E}$ from Definition 8.1 has probability at least $1-\delta$.*

Proof. Consider a fixed $i \in [M]$ and t and write

$$\begin{aligned}\widehat{u}_t^i - u_t^i &= \sum_{\ell=1}^{t} \mathbf{1}\{i_\ell = i\}\,(r_t - v^{\pi_t}) \\ &= \sum_{\ell=1}^{t} \mathbf{1}\{i_\ell = i\}\,(r_\ell - \mathbb{E}[r_\ell|\pi_\ell])\end{aligned}$$

Let $\mathcal{F}_t$ be the sigma-field induced by all variables up to round t before the reward is revealed, i.e., $\mathcal{F}_t = \sigma\left(\{x_\ell, \pi_\ell, i_\ell\}_{\ell\in[t-1]} \cup \{x_t, \pi_t, t_t\}\right)$. Then, $X_\ell = \mathbf{1}\{i_\ell = i\}\,(r_t - \mathbb{E}[r_t|\pi_t]) \in [-1, +1]$ is a martingale-difference sequence w.r.t. $\mathcal{F}_\ell$. We will now apply a Hoeffding-style uniform concentration bound from Howard et al. (2018). Using the terminology and definition in this article, by case Hoeffding I in Table 4, the process $S_t = \sum_{\ell=1}^t X_\ell$ is sub-ψ_N with variance process $V_t = \sum_{\ell=1}^t \mathbf{1}\{i_\ell = i\}/4$. Thus by using the boundary choice in Equation (11) of Howard et al. (2018), we get

$$S_t \leq 1.7\sqrt{V_t\left(\ln\ln(2V_t) + 0.72\ln(5.2/\delta)\right)} = 0.85\sqrt{n_t^i\left(\ln\ln(n_t^i/2) + 0.72\ln(5.2/\delta)\right)}$$

for all k where $V_k \geq 1$ with probability at least $1-\delta$. Applying the same argument to $-S_k$ gives that

$$\left|\widehat{u}_t^i - u_t^i\right| \leq 3 \vee 0.85\sqrt{n_t^i\left(\ln\ln(n_t^i/2) + 0.72\ln(10.4/\delta)\right)}$$

holds with probability at least $1-\delta$ for all t.

We now take a union bound over $i \in [M]$ and rebind $\delta \to \delta/M$. Then picking the absolute constant c sufficiently large gives the desired statement. ☐

Lemma 8.3 (Balancing potential lemma)**.** *For each $i \in [M]$, let $F_i : \mathbb{N}\cup\{0\} \to \mathbb{R}_+$ be a nondecreasing potential function that does not increase too quickly, i.e.,*

$$F_i(\ell) \leq F_i(\ell+1) \leq \alpha \cdot F_i(\ell) \qquad \forall \ell \in \mathbb{N}\cup\{0\}$$

and that $0 < F_i(0) \leq \alpha \cdot F_j(0)$ for all $i, j \in [M]^2$. Consider a sequence $(i_t)_{t\in\mathbb{N}}$ such that $i_t = \operatorname{argmin}_{i\in[M]} F_i(n_{t-1}^i)$ and $n_t^i = \sum_{\ell=1}^t \mathbf{1}\{i_\ell = i\}$, i.e., $i_t \in [M]$ is always chosen as the smallest current potential. Then, for all $t \in \mathbb{N}$

$$\max_{i\in[M]} F_i(n_t^i) \leq \alpha \cdot \min_{j\in[M]} F_j(n_t^j).$$

Proof. Our proof works by induction over t. At $t = 1$, we have $n_0^i = 0$ for all $i \in [M]$ and thus, by assumption, the statement holds. Assume now the statement holds for t. Notice that since n_t^i and F_i are non-decreasing, we have for all $i \in [M]$

$$\min_i F_i(n_t^i) \geq \min_i F_i(n_{t-1}^i)).$$

Further, for all $i \neq i_t$ that were not chosen in round t, we even have $F_i(n_{t-1}^i) = F_i(n_t^i)$ for all $i \neq i_t$. We now distinguish two cases:

Case $i_t \notin \operatorname{argmax}_i F_i(n_{t-1}^i)$**.** Since the potential of all $i \neq i_t$ that attain the max is unchanged, we have

$$\max_i F_i(n_t^i) = \max_i F_i(n_{t-1}^i)$$

and therefore $\frac{\max_i F_i(n_t^i)}{\min_j F_j(n_t^j)} \leq \frac{\max_i F_i(n_{t-1}^i)}{\min_j F_j(n_{t-1}^j)} \leq \alpha$.

Case $i_t \in \operatorname{argmax}_i F_i(n_{t-1}^i)$**.** Since i_t attains both the maximum and the minimum, and hence all potentials are identical, we have

$$\max_i F_i(n_t^i) = F_{i_t}(n_t^{i_t}) \leq F_{i_t}(n_{t-1}^{i_t} + 1) \leq \alpha F_{i_t}(n_{t-1}^{i_t}) = \alpha \min_j F_j(n_{t-1}^j) \ .$$

This concludes the proof. ☐

9 PROOFS FOR THE DOUBLING ALGORITHM (ALGORITHM 1, LEFT)

Lemma 9.1. *In event $\mathcal{E}$, for each base learner i all rounds $t \in \mathbb{N}$, the regret multiplier $\widehat{d}_t^i$ satisfies*

$$\widehat{d}_t^i \leq 2\bar{d}_t^i \ .$$

Proof. Note that instead of showing this for all rounds t, we can also show this equivalently for all number k of plays of base learner i. If the statement is violated for base learner i, then there is a minimum number k of plays at which this statement is violated. Note that by definition $\bar{d}_{(0)}^i = d_{\min}$ and by initialization $\widehat{d}_{(0)}^i = d_{\min}$, hence this k cannot be 0.

Consider now the round t where the learner i was played the k-th time, i.e., the first round at which the statement was violated. This means $\widehat{d}_t^i > 2\bar{d}_t^i$ but $\widehat{d}_{t-1}^i \leq \bar{2}d_{t-1}^i$ still holds. Since $\widehat{d}_t^i$ can be at most $2\widehat{d}_{t-1}^i$, we have $\widehat{d}_{t-1}^i > \bar{d}_t^i$. We will now show that in this case, the misspecification test could not have triggered and therefore $\widehat{d}_t^i = \widehat{d}_{t-1}^i \leq 2\bar{d}_{t-1}^i \leq 2\bar{d}_t^i$ which is a contradiction. To show that the test cannot trigger, consider the LHS of the test condition and bound it from below as

$$\begin{aligned}
\frac{\widehat{u}_t^{i_t}}{n_t^{i_t}} + \frac{\widehat{d}_{t-1}^{i_t}\sqrt{n_t^{i_t}}}{n_t^{i_t}} + c\sqrt{\frac{\ln\frac{M\ln n_t^{i_t}}{\delta}}{n_t^{i_t}}} &\geq \frac{u_t^{i_t}}{n_t^{i_t}} + \frac{\widehat{d}_{t-1}^{i_t}\sqrt{n_t^{i_t}}}{n_t^{i_t}} && \text{(Event } \mathcal{E}\text{)}\\
&\geq \frac{u_t^{i_t}}{n_t^{i_t}} + \frac{\bar{d}_t^{i_t}\sqrt{n_t^{i_t}}}{n_t^{i_t}} && (\widehat{d}_{t-1}^{i_t} > \bar{d}_t^{i_t})\\
&\geq \frac{u_t^{i_t} + \sum_{\ell=1}^{n_t^{i_t}} \mathrm{reg}(\pi_{(\ell)}^{i_t})}{n_t^{i_t}} && \text{(definition of } d_t^i\text{)}\\
&\geq v^\star && \text{(definition of regret)}\\
&\geq \frac{u_t^j}{n_t^j} && \text{(definition of } v^\star\text{)}\\
&\geq \frac{u_t^j}{n_t^j} - c\sqrt{\frac{\ln\frac{M\ln n_t^j}{\delta}}{n_t^j}}. && \text{(Event } \mathcal{E}\text{)}
\end{aligned}$$

This holds for any $j \in [M]$ and thus, the test does not trigger. ☐

Corollary 9.1. *In event $\mathcal{E}$, for each base learner i all rounds $t \in \mathbb{N}$, the number of times the regret multiplier $\widehat{d}_t^i$ has doubled so far is bounded as follows:*

$$\widehat{d}_t^i \leq 1 + \log_2 \frac{\bar{d}_t^i}{d_{\min}} .$$

Lemma 9.2. *The potentials in Algorithm 1 (left) are balanced at all times up to a factor 3, that is, $\phi_t^i \leq 3\phi_t^j$ for all rounds $t \in \mathbb{N}$ and base learners $i, j \in [M]$.*

Proof. We will show that Lemma 8.3 with $\alpha = 3$ holds when we apply the lemma to $F_i(n_{t-1}^i) = \phi_t^i$.

First $F_i(0) = \phi_1^i = d_{\min}$ for all $i \in [M]$ and, thus, the initial condition holds. To show the remaining condition, it suffices to show that ϕ_t^i is non-decreasing in t and cannot increase more than a factor of 3 per round. If i was not played in round t, then $\phi_t^i = \phi_{t-1}^i$ and both conditions holds. If i was played, i.e., $i = i_t$, then

$$\phi_t^i = \widehat{d}_t^i \sqrt{n_t^i} \leq 2\widehat{d}_{t-1}^i \sqrt{n_t^i} \leq \begin{cases} 2\widehat{d}_{t-1}^i \sqrt{n_t^i - 1} \sqrt{\frac{n_t^i}{n_t^i - 1}} = 2\phi_{t-1}^i \sqrt{\frac{n_t^i}{n_t^i - 1}} \leq 3\phi_{t-1}^i & \text{if } n_t^i > 1 \\ 2\widehat{d}_{\min} \sqrt{1} = \phi_{t-1}^i & \text{if } n_t^i = 1 \end{cases},$$

as claimed. ☐

Lemma 9.3. *In event $\mathcal{E}$, the regret of all base learners i is bounded in all rounds T as*

$$\sum_{k=1}^{n_T^i} \mathrm{reg}(\pi_{(k)}^i) \leq \frac{6(\bar{d}_T^j)^2}{d_{\min}} \sqrt{n_T^i} + 6\bar{d}_T^j \sqrt{n_T^j} + \left(6c\frac{\bar{d}_T^j}{d_{\min}} + 2c\right) \sqrt{n_T^i \ln \frac{M \ln T}{\delta}} + 1 + \log_2 \frac{\bar{d}_T^i}{d_{\min}} ,$$

where $j \in [M]$ is an arbitrary base learner with $n_T^j > 0$.

Proof. Consider a fixed base learner i and time horizon T, and let $t \leq T$ be the last round where i was played but the misspecification test did not trigger. If no such round exists, then set $t = 0$. By Corollary 9.1, i can be played at most $1 + \log_2 \frac{\bar{d}_T^i}{d_{\min}}$ times between t and T and thus

$$\sum_{k=1}^{n_T^i} \mathrm{reg}(\pi_{(k)}^i) \leq \sum_{k=1}^{n_t^i} \mathrm{reg}(\pi_{(k)}^i) + 1 + \log_2 \frac{\bar{d}_T^i}{d_{\min}}.$$

If $t = 0$, then the desired statement holds. Thus, it remains to bound the first term in the RHS above when $t > 0$. Since $i = i_t$ and the test did not trigger we have, for any base learner j with $n_t^j > 0$,

$$\begin{aligned}
\sum_{k=1}^{n_t^i} \mathrm{reg}(\pi_{(k)}^i) &= n_t^i v^\star - u_t^i && \text{(definition of regret)} \\
&= n_t^i v^\star - \frac{n_t^i}{n_t^j} u_t^j + \frac{n_t^i}{n_t^j} u_t^j - u_t^i \\
&= \frac{n_t^i}{n_t^j} \left(n_t^j v^\star - u_t^j \right) + \frac{n_t^i}{n_t^j} u_t^j - u_t^i \\
&= \frac{n_t^i}{n_t^j} \left(\sum_{k=1}^{n_t^j} \mathrm{reg}(\pi_{(k)}^j) \right) + \frac{n_t^i}{n_t^j} u_t^j - u_t^i && \text{(definition of regret)} \\
&\leq \frac{n_t^i}{n_t^j} \left(d_t^j \sqrt{n_t^j} \right) + \frac{n_t^i}{n_t^j} u_t^j - u_t^i && \text{(definition of regret rate)} \\
&\leq \sqrt{\frac{n_t^i}{n_t^j}} d_t^j \sqrt{n_t^i} + \frac{n_t^i}{n_t^j} u_t^j - u_t^i.
\end{aligned}$$

We now use the balancing condition in Lemma 9.2 to bound the first factor $\sqrt{n_t^i/n_t^j}$. This condition gives that $\phi_{t+1}^i \le 3\phi_{t+1}^j$. Since both $n_t^j > 0$ and $n_t^i > 0$, we have $\phi_{t+1}^i = \widehat{d}_t^i\sqrt{n_t^i}$ and $\phi_{t+1}^j = \widehat{d}_t^j\sqrt{n_t^j}$. Thus, we get

$$\sqrt{\frac{n_t^i}{n_t^j}} = \sqrt{\frac{n_t^i}{n_t^j}} \cdot \frac{\widehat{d}_t^i}{\widehat{d}_t^j} \cdot \frac{\widehat{d}_t^j}{\widehat{d}_t^i} = \frac{\phi_{t+1}^i}{\phi_{t+1}^j} \cdot \frac{\widehat{d}_t^j}{\widehat{d}_t^i} \le 3\frac{\widehat{d}_t^j}{\widehat{d}_t^i} \le 6\frac{\bar{d}_t^j}{d_{\min}}. \tag{2}$$

Plugging this back into the expression above, we have

$$\sum_{k=1}^{n_t^i} \mathrm{reg}(\pi_{(k)}^i) \le \frac{6(\bar{d}_t^j)^2}{d_{\min}}\sqrt{n_t^i} + \frac{n_t^i}{n_t^j}u_t^j - u_t^i.$$

To bound the last two terms, we use the fact that the misspecification test did not trigger in round t. Therefore

$$\begin{aligned}
u_t^i &\ge \widehat{u}_t^i - c\sqrt{n_t^i \ln\frac{M\ln n_t^i}{\delta}} && \text{(event } \mathcal{E}\text{)}\\
&= n_t^i\left(\frac{\widehat{u}_t^i}{n_t^i} + c\sqrt{\frac{\ln\frac{M\ln n_t^i}{\delta}}{n_t^i}} + \frac{\widehat{d}_t^i}{\sqrt{n_t^i}}\right) - 2c\sqrt{n_t^i\ln\frac{M\ln n_t^i}{\delta}} - \widehat{d}_t^i\sqrt{n_t^i}\\
&\ge \frac{n_t^i}{n_t^j}\widehat{u}_t^j - \sqrt{\frac{n_t^i}{n_t^j}}c\sqrt{n_t^i\ln\frac{M\ln n_t^j}{\delta}} - 2c\sqrt{n_t^i\ln\frac{M\ln n_t^i}{\delta}} - \widehat{d}_t^i\sqrt{n_t^i} && \text{(test not triggered)}
\end{aligned}$$

Rearranging terms and plugging this expression in the bound above gives

$$\begin{aligned}
\sum_{k=1}^{n_t^i} \mathrm{reg}(\pi_{(k)}^i) &\le \frac{6(\bar{d}_t^j)^2}{d_{\min}}\sqrt{n_t^i} + \sqrt{\frac{n_t^i}{n_t^j}}c\sqrt{n_t^i\ln\frac{M\ln n_t^j}{\delta}} + 2c\sqrt{n_t^i\ln\frac{M\ln n_t^i}{\delta}} + \widehat{d}_t^i\sqrt{n_t^i}\\
&\le \frac{6(\bar{d}_t^j)^2}{d_{\min}}\sqrt{n_t^i} + 6\frac{\bar{d}_t^j}{d_{\min}}c\sqrt{n_t^i\ln\frac{M\ln n_t^j}{\delta}} + 2c\sqrt{n_t^i\ln\frac{M\ln n_t^i}{\delta}} + \widehat{d}_t^i\sqrt{n_t^i} && \text{(Equation 2)}\\
&\le \frac{6(\bar{d}_t^j)^2}{d_{\min}}\sqrt{n_t^i} + 6\frac{\bar{d}_t^j}{d_{\min}}c\sqrt{n_t^i\ln\frac{M\ln n_t^j}{\delta}} + 2c\sqrt{n_t^i\ln\frac{M\ln n_t^i}{\delta}} + 3\widehat{d}_t^j\sqrt{n_t^j} && \text{(Equation 2)}\\
&\le \frac{6(\bar{d}_t^j)^2}{d_{\min}}\sqrt{n_t^i} + 3\widehat{d}_t^j\sqrt{n_t^j} + \left(6c\frac{\bar{d}_t^j}{d_{\min}} + 2c\right)\sqrt{n_t^i\ln\frac{M\ln t}{\delta}} && (n_t^i \le t)\\
&\le \frac{6(\bar{d}_t^j)^2}{d_{\min}}\sqrt{n_t^i} + 6\bar{d}_t^j\sqrt{n_t^j} + \left(6c\frac{\bar{d}_t^j}{d_{\min}} + 2c\right)\sqrt{n_t^i\ln\frac{M\ln t}{\delta}} && \text{(Lemma 9.1)}
\end{aligned}$$

Finally, since $t \le T$ and therefore $\bar{d}_t^j \le \bar{d}_T^j$ and $n_t^j \le n_T^j$ (and similarly for i), the statement follows. □

Theorem 3.1. *With probability at least $1-\delta$, the regret of D^3RB (Algorithm 1, left) with parameters δ and $d_{\min} \ge 1$ is bounded in all rounds $T \in \mathbb{N}$ as*[3]

$$\mathrm{Reg}(T) = \tilde{O}\left(\bar{d}_T^\star M\sqrt{T} + (\bar{d}_T^\star)^2\sqrt{MT}\right)$$

where $\bar{d}_T^\star = \min_{i\in[M]} \bar{d}_T^i = \min_{i\in[M]}\max_{t\in[T]} d_t^i$ is the smallest monotonic regret coefficient among all learners (see Definition 2.1).

Proof. By Lemma 8.2, event $\mathcal{E}$ from Definition 8.1 has probability at least $1-\delta$. In event $\mathcal{E}$, we can apply Lemma 9.3 for each base learner. Summing up the bound from that lemma gives

$$\mathrm{Reg}(T) \le \sum_{i=1}^{M}\left[\frac{6(\bar{d}_T^j)^2}{d_{\min}}\sqrt{n_T^i} + 6\bar{d}_T^j\sqrt{n_T^j} + \left(6c\frac{\bar{d}_T^j}{d_{\min}} + 2c\right)\sqrt{n_T^i\ln\frac{M\ln T}{\delta}} + 1 + \log_2\frac{\bar{d}_T^i}{d_{\min}}\right]$$

[3] Here and throughout, $\tilde{O}$ hides log-factors.

$$\leq 6M\bar{d}_T^j\sqrt{T} + M + M\log_2\frac{\sqrt{T}}{d_{\min}} + \left[\frac{6(\bar{d}_T^j)^2}{d_{\min}} + \frac{4\bar{d}_T^j}{d_{\min}}2c\sqrt{\ln\frac{M\ln T}{\delta}}\right]\sum_{i=1}^{M}\sqrt{n_T^i}$$

$$\leq \left(6\sqrt{M}\bar{d}_T^j + \frac{6(\bar{d}_T^j)^2}{d_{\min}} + \frac{8c\bar{d}_T^j}{d_{\min}}\sqrt{\ln\frac{M\ln T}{\delta}}\right)\sqrt{MT} + M + M\log_2\frac{T}{d_{\min}}.$$

Plugging in $d_{\min} \geq 1$ yields

$$\begin{aligned}
\mathrm{Reg}(T) &\leq \left(6\sqrt{M}\bar{d}_T^j + 6(\bar{d}_T^j)^2 + 8c\bar{d}_T^j\sqrt{\ln\frac{M\ln T}{\delta}}\right)\sqrt{MT} + M + M\log_2 T \\
&= O\left(\left(M\bar{d}_T^j + \sqrt{M}(\bar{d}_T^j)^2 + \bar{d}_T^j\sqrt{\ln\frac{M\ln T}{\delta}}\right)\sqrt{T} + M\ln(T)\right) \\
&= \tilde{O}\left(\bar{d}_T^j M\sqrt{T} + (\bar{d}_T^j)^2\sqrt{MT}\right) ,
\end{aligned}$$

as desired. □

10 PROOFS FOR THE ESTIMATING ALGORITHM (ALGORITHM 1, RIGHT)

Lemma 10.1. *In event $\mathcal{E}$, the regret rate estimate in Algorithm 1 (right) does not overestimate the current regret rate, that is, for all base learners $i \in [M]$ and rounds $t \in \mathbb{N}$, we have*

$$\widehat{d}_t^i \leq d_t^i.$$

Proof. Note that the algorithm only updates $\widehat{d}_t^i$ when learner i is chosen and only then d_t^i changes. Further, the condition holds initially since $\widehat{d}_1^i = d_{\min} \leq d_t^i$. Hence, it is sufficient to show that this condition holds whenever $\widehat{d}_t^i$ is updated. The algorithm estimates $\widehat{d}_t^i$ as

$$\widehat{d}_t^i = \max\left\{d_{\min},\ \sqrt{n_t^i}\left(\max_{j\in[M]}\frac{\hat{u}_t^j}{n_t^j} - c\sqrt{\frac{\ln\frac{M\ln n_t^j}{\delta}}{n_t^j}} - \frac{\hat{u}_t^{i_t}}{n_t^i} - c\sqrt{\frac{\ln\frac{M\ln n_t^i}{\delta}}{n_t^i}}\right)\right\} .$$

If $\widehat{d}_t^i \leq d_{\min}$, then the result holds since by definition $d_t^i \geq d_{\min}$. In the other case, we have

$$\begin{aligned}
\widehat{d}_t^i &= \sqrt{n_t^i}\left(\max_{j\in[M]}\frac{\hat{u}_t^j}{n_t^j} - c\sqrt{\frac{\ln\frac{M\ln n_t^j}{\delta}}{n_t^j}} - \frac{\hat{u}_t^i}{n_t^i} - c\sqrt{\frac{\ln\frac{M\ln n_t^i}{\delta}}{n_t^i}}\right) \\
&\leq \sqrt{n_t^i}\left(\max_{j\in[M]}\frac{u_t^j}{n_t^j} - \frac{u_t^i}{n_t^i}\right) && \text{(event } \mathcal{E}) \\
&\leq \sqrt{n_t^i}\left(v^\star - \frac{u_t^{i_t}}{n_t^i}\right) && \text{(definition of optimal value } v^\star) \\
&= \frac{n_t^i v^\star - u_t^i}{\sqrt{n_t^i}} = \frac{\sum_{k=1}^{n_t^i}\mathrm{reg}(\pi_{(k)}^i)}{\sqrt{n_t^i}} && \text{(regret definition)} \\
&\leq d_t^i , && \text{(definition of } d_t^i)
\end{aligned}$$

as claimed. □

Lemma 10.2. *In event $\mathcal{E}$, the balancing potentials ϕ_t^i in Algorithm 1 (right) satisfy for all $t \in \mathbb{N}$ and $i \in [M]$ where $n_t^i \geq 1$*

$$\phi_{t+1}^i \leq d_t^i\sqrt{n_t^i}.$$

Proof. If $i \neq i_t$, then $\phi_{t+1}^i = \phi_t^i$, $d_t^i = d_{t-1}^i$ and $n_t^i = n_{t-1}^i$. It is therefore sufficient to only check this condition for $i = i_t$. By definition of the balancing potential, we have when $i = i_t$

$$\phi_{t+1}^i \leq \max\left\{\phi_t^i, \widehat{d}_t^i\sqrt{n_t^i}\right\} \leq \max\left\{\phi_t^i, d_t^i\sqrt{n_t^i}\right\} \;,$$

where the last inequality holds because of Lemma 10.1. If $n_t^i = 1$, then $\phi_t^i = d_{\min}$ and $d_t^i\sqrt{n_t^i} \geq d_{\min}\sqrt{1}$ by definition, and the statement holds. Otherwise, we can assume that $\phi_t^i \leq d_{t-1}^i\sqrt{n_{t-1}^i}$ by induction. This gives

$$\phi_{t+1}^i \leq \max\left\{d_{t-1}^i\sqrt{n_{t-1}^i}, d_t^i\sqrt{n_t^i}\right\}.$$

We notice that $d_t^i\sqrt{n_t^i} = \max\{d_{\min}\sqrt{n_t^i}, \sum_{k=1}^{n_t^i} \mathrm{reg}(\pi_{(k)}^i)\}$. Since each term inside the max is non-decreasing in t, $d_t^i\sqrt{n_t^i}$ is also non-decreasing in t, and therefore $\phi_{t+1}^i \leq d_t^i\sqrt{n_t^i}$, as anticipated. □

Lemma 10.3. *In event $\mathcal{E}$, for all $T \in \mathbb{N}$ and $i \in [M]$, the number of times the balancing potential ϕ_t^i doubled until time T in Algorithm 1 (right) is bounded by*

$$\log_2\left(t\max\{1, 1/d_{\min}\}\right).$$

Proof. The balancing potential ϕ_t^i is non-decreasing in t and $\phi_1^i = d_{\min}$. Further, by Lemma 10.2, we have

$$\phi_{t+1}^i \leq d_t^i\sqrt{n_t^i} \leq \max\left\{d_{\min}\sqrt{n_t^i}, n_t^i\right\}.$$

Thus, the number of times ϕ_t^i can double is at most

$$\log_2\left(\max\left\{\sqrt{n_t^i}, \frac{n_t^i}{d_{\min}}\right\}\right) \leq \log_2\left(t\max\{1, 1/d_{\min}\}\right) \;.$$

This concludes the proof. □

Lemma 10.4. *The balancing potentials in Algorithm 1 (right) are balanced at all times up to a factor 2, that is, $\phi_t^i \leq 2\phi_t^j$ for all rounds $t \in \mathbb{N}$ and base learners $i, j \in [M]$.*

Proof. We will show that Lemma 8.3 with $\alpha = 2$ holds when we apply the lemma to $F_i(n_{t-1}^i) = \phi_t^i$.

First $F_i(0) = \phi_1^i = d_{\min}$ for all $i \in [M]$ and, thus, the initial condition holds. To show the remaining condition, it suffices to show that ϕ_t^i is non-decreasing in t and cannot increase more than a factor of 2 per round. This holds by the clipping in the definition of ϕ_{t+1}^i in the algorithm. □

Lemma 10.5. *In event $\mathcal{E}$, the regret of all base learners i is bounded in all rounds T as*

$$\sum_{k=1}^{n_T^i} \mathrm{reg}(\pi_{(k)}^i) \leq \frac{2(d_t^j)^2}{d_{\min}}\sqrt{n_t^i} + 2d_t^j\sqrt{n_t^j} + 2c\left(1 + \frac{2d_t^j}{d_{\min}}\right)\sqrt{n_t^i \ln\frac{M\ln t}{\delta}} + \log_2\max\left\{T, \frac{T}{d_{\min}}\right\},$$

where $j \in [M]$ is an arbitrary base learner with $n_T^j > 0$ and $t \leq T$ is the last round where $i = i_t$ and $\phi_{t+1}^i < 2\phi_t^i$.

Proof. Consider fixed base learner i and time horizon T, and let $t \leq T$ be the last round where i was played and ϕ_t^i did not double, i.e., $\phi_{t+1}^i < 2\phi_t^i$. If no such round exists, then set $t = 0$. By Lemma 10.3, i can be played at most $\log_2\left(T\max\{1, 1/d_{\min}\}\right)$ times between t and T and thus

$$\sum_{k=1}^{n_T^i} \mathrm{reg}(\pi_{(k)}^i) \leq \sum_{k=1}^{n_t^i} \mathrm{reg}(\pi_{(k)}^i) + \log_2\left(T\max\{1, 1/d_{\min}\}\right).$$

If $t = 0$, then the desired statement holds. Thus, it remains to bound the first term above when $t > 0$. We can write the regret of base learner i up to t in terms of the regret of any learner j with $n_t^j > 0$ as follows:

$$\begin{aligned}
\sum_{k=1}^{n_t^i} \mathrm{reg}(\pi_{(k)}^i) &= n_t^i v^\star - u_t^i && \text{(definition of regret)} \\
&= n_t^i v^\star - \frac{n_t^i}{n_t^j} u_t^j + \frac{n_t^i}{n_t^j} u_t^j - u_t^i \\
&= \frac{n_t^i}{n_t^j}\left(n_t^j v^\star - u_t^j\right) + \frac{n_t^i}{n_t^j} u_t^j - u_t^i \\
&= \frac{n_t^i}{n_t^j}\left(\sum_{k=1}^{n_t^j} \mathrm{reg}(\pi_{(k)}^j)\right) + \frac{n_t^i}{n_t^j} u_t^j - u_t^i && \text{(definition of regret)} \\
&\le \frac{n_t^i}{n_t^j}\left(d_t^j \sqrt{n_t^j}\right) + \frac{n_t^i}{n_t^j} u_t^j - u_t^i && \text{(definition of regret rate)} \\
&\le \sqrt{\frac{n_t^i}{n_t^j}} d_t^j \sqrt{n_t^i} + \frac{n_t^i}{n_t^j} u_t^j - u_t^i.
\end{aligned}$$

We now use the balancing condition in Lemma 10.4 to bound the first factor $\sqrt{n_t^i/n_t^j}$. This condition gives that $\phi_{t+1}^i \le 2\phi_{t+1}^j$. Since $\phi_{t+1}^i < 2\phi_t^i$ and, thus, the balancing potential was not clipped from above, we have $\phi_{t+1}^i \ge \widehat{d}_t^i \sqrt{n_t^i}$. Further, since $n_t^j > 0$ we can apply Lemma 10.2 to get $\phi_{t+1}^j \le d_t^j \sqrt{n_t^j}$. Thus, we get

$$\sqrt{\frac{n_t^i}{n_t^j}} = \sqrt{\frac{n_t^i}{n_t^j}} \cdot \frac{\widehat{d}_t^i}{d_t^j} \cdot \frac{d_t^j}{\widehat{d}_t^i} \le \frac{\phi_{t+1}^i}{\phi_{t+1}^j} \cdot \frac{d_t^j}{\widehat{d}_t^i} \le 2\frac{d_t^j}{\widehat{d}_t^i} \le 2\frac{d_t^j}{d_{\min}}. \tag{3}$$

Plugging this back into the expression above, we have

$$\sum_{k=1}^{n_t^i} \mathrm{reg}(\pi_{(k)}^i) \le \frac{2(d_t^j)^2}{d_{\min}} \sqrt{n_t^i} + \frac{n_t^i}{n_t^j} u_t^j - u_t^i.$$

To bound the last two terms, we use the regret coefficient estimate:

$$\begin{aligned}
\frac{n_t^i}{n_t^j} u_t^j - u_t^i &= n_t^i \left(\frac{u_t^j}{n_t^j} - \frac{u_t^i}{n_t^i}\right) \\
&\le n_t^i \left(\frac{\hat{u}_t^j}{n_t^j} - \frac{\hat{u}_t^i}{n_t^i}\right) + c\sqrt{n_t^i \ln \frac{M \ln n_t^i}{\delta}} + c n_t^i \sqrt{\frac{\ln \frac{M \ln n_t^j}{\delta}}{n_t^j}} && \text{(event } \mathcal{E}) \\
&= n_t^i \left(\frac{\hat{u}_t^j}{n_t^j} - c\sqrt{\frac{\ln \frac{M \ln n_t^j}{\delta}}{n_t^j}} - \frac{\hat{u}_t^i}{n_t^i} - c\sqrt{\frac{\ln \frac{M \ln n_t^i}{\delta}}{n_t^i}}\right) + 2c\sqrt{n_t^i \ln \frac{M \ln n_t^i}{\delta}} + 2c n_t^i \sqrt{\frac{\ln \frac{M \ln n_t^j}{\delta}}{n_t^j}} \\
&\le \widehat{d}_t^i \sqrt{n_t^i} + 2c\sqrt{n_t^i \ln \frac{M \ln n_t^i}{\delta}} + 2c n_t^i \sqrt{\frac{\ln \frac{M \ln n_t^j}{\delta}}{n_t^j}} && \text{(definition of } \widehat{d}_t^i) \\
&\le \widehat{d}_t^i \sqrt{n_t^i} + 2c\left(1 + \sqrt{\frac{n_t^i}{n_t^j}}\right)\sqrt{n_t^i \ln \frac{M \ln t}{\delta}} && (n_t^i \le t \text{ and } n_t^j \le t) \\
&\le \widehat{d}_t^i \sqrt{n_t^i} + 2c\left(1 + 2\frac{d_t^j}{d_{\min}}\right)\sqrt{n_t^i \ln \frac{M \ln t}{\delta}} && \text{(Equation 3)} \\
&\le \phi_{t+1}^i + 2c\left(1 + 2\frac{d_t^j}{d_{\min}}\right)\sqrt{n_t^i \ln \frac{M \ln t}{\delta}} && (\phi_{t+1}^i \ge \widehat{d}_t^i \sqrt{n_t^i})
\end{aligned}$$

$$\leq 2\phi_{t+1}^j + 2c\left(1 + 2\frac{d_t^j}{d_{\min}}\right)\sqrt{n_t^i \ln \frac{M \ln t}{\delta}} \qquad \text{(Lemma 10.4)}$$

$$\leq 2d_t^j\sqrt{n_t^j} + 2c\left(1 + 2\frac{d_t^j}{d_{\min}}\right)\sqrt{n_t^i \ln \frac{M \ln t}{\delta}}. \qquad \text{(Lemma 10.2)}$$

Plugging this back into the expression above, we get

$$\sum_{k=1}^{n_T^i} \mathrm{reg}(\pi_{(k)}^i) \leq \frac{2(d_t^j)^2}{d_{\min}}\sqrt{n_t^i} + 2d_t^j\sqrt{n_t^j} + 2c\left(1 + \frac{2d_t^j}{d_{\min}}\right)\sqrt{n_t^i \ln \frac{M \ln t}{\delta}} + \log_2 \max\left\{T, \frac{T}{d_{\min}}\right\} ,$$

which is the desired statement. □

Theorem 3.2. *With probability at least $1-\delta$, the regret of ED^{2}RB (Algorithm 1, right) with parameters δ and $d_{\min} \geq 1$ is bounded in all rounds $T \in \mathbb{N}$ as*

$$\mathrm{Reg}(T) = \tilde{O}\left(d_T^\star M\sqrt{T} + (d_T^\star)^2\sqrt{MT}\right)$$

where $d_T^\star = \min_{i\in[M]} \max_{j\in[M]} d_{T_j}^i$ is the smallest regret coefficient among all learners, and T_j is the last time t when base learner j was played and $\phi_{t+1}^j < 2\phi_t^j$.

Proof. By Lemma 8.2, event $\mathcal{E}$ from Definition 8.1 has probability at least $1-\delta$. In event $\mathcal{E}$, we can apply Lemma 10.5 for each base learner. Summing up the bound for all base learners $i \in [M]$ with $j \in \mathrm{argmin}_{i' \in [M]} \max_i d_{T_i}^{i'}$ from that lemma gives

$$\begin{aligned}
\mathrm{Reg}(T) &\leq \sum_{i=1}^{M}\left[\frac{2(d_{T_i}^j)^2}{d_{\min}}\sqrt{n_{T_i}^i} + 2d_{T_i}^j\sqrt{n_{T_i}^j} + 2c\left(1 + \frac{2d_{T_i}^j}{d_{\min}}\right)\sqrt{n_{T_i}^i \ln \frac{M \ln T}{\delta}} + \log_2 \max\left\{T, \frac{T}{d_{\min}}\right\}\right] \\
&\leq 2Md_T^\star\sqrt{T} + M\log_2 \max\left\{T, \frac{T}{d_{\min}}\right\} + \left[\frac{2(d_T^\star)^2}{d_{\min}} + \frac{6d_T^\star}{d_{\min}}c\sqrt{\ln \frac{M \ln T}{\delta}}\right]\sum_{i=1}^{M}\sqrt{n_T^i} \\
&\leq \left(2\sqrt{M}d_T^\star + \frac{2(d_T^\star)^2}{d_{\min}} + \frac{6cd_T^\star}{d_{\min}}\sqrt{\ln \frac{M \ln T}{\delta}}\right)\sqrt{MT} + M\log_2 \max\left\{T, \frac{T}{d_{\min}}\right\}.
\end{aligned}$$

Here we have used that $d_{T_i}^j \leq d_T^\star$ for all $i \in [M]$ by the definition of $d_T^\star$. Plugging in $d_{\min} \geq 1$ gives

$$\begin{aligned}
\mathrm{Reg}(T) &\leq \left(2\sqrt{M}d_T^\star + 2(d_T^\star)^2 + 6cd_T^\star\sqrt{\ln \frac{M \ln T}{\delta}}\right)\sqrt{MT} + M\log_2 T \\
&= O\left(\left(Md_T^\star + \sqrt{M}(d_T^\star)^2 + d_T^\star\sqrt{M \ln \frac{M \ln T}{\delta}}\right)\sqrt{T} + M\ln(T)\right) \\
&= \tilde{O}\left(d_T^\star M\sqrt{T} + (d_T^\star)^2\sqrt{MT}\right) ,
\end{aligned}$$

as claimed. □

11 EXPERIMENTAL DETAILS

We used a 50 core machine to run our experiments. We made use of this computing infrastructure by parallelizing our experiment runs. The experiments take 12 hours to complete.

11.1 Meta-Learners

We now list the meta-learners used in our experiments.

Algorithm 2: CORRAL Meta-Algorithm

Input: M base learners, learning rate η.
Initialize: $\gamma = 1/T, \beta = e^{\frac{1}{\ln T}}, \eta_{1,j} = \eta, \rho_1^j = 2M, \underline{p}_1^j = \frac{1}{\rho_1^j}, p_1^j = 1/M$ for all $j \in [M]$.
for $t = 1, \cdots, T$ **do**
  Sample $i_t \sim p_t$.
  Receive feedback r_t from base learner i_t.
  Update p_t, η_t, $\underline{p}_t$ and ρ_t to p_{t+1}, η_{t+1}, $\underline{p}_{t+1}$ and ρ_{t+1} using $\mathsf{CORRAL-Update}$ Algorithm 4.

Algorithm 3: Log-Barrier-OMD(p_t, ℓ_t, η_t)

Input: learning rate vector η_t, previous distribution p_t and current loss ℓ_t
Output: updated distribution p_{t+1}
Find $\lambda \in [\min_j \ell_{t,j}, \max_j \ell_{t,j}]$ such that $\sum_{j=1}^M \frac{1}{\frac{1}{p_t^i} + \eta_{t,j}(\ell_{t,j} - \lambda)} = 1$
Return p_{t+1} such that $\frac{1}{p_{t+1}^j} = \frac{1}{p_t^j} + \eta_{t,j}(\ell_{t,j} - \lambda)$

Algorithm 4: $\mathsf{CORRAL-Update}$

Input: learning rate vector η_t, distribution p_t, lower bound $\underline{p}_t$ and current loss r_t
Output: updated distribution p_{t+1}, learning rate η_{t+1} and loss range ρ_{t+1}
Update $p_{t+1} = $ Log-Barrier-OMD$(p_t, \frac{r_t}{p_{t,j_t}}\mathbf{e}_{j_t}, \eta_t)$.
Set $p_{t+1} = (1-\gamma)p_{t+1} + \gamma\frac{1}{M}$.
for $j = 1, \cdots, M$ **do**
  if $\underline{p}_t^j > p_{t+1}^j$ **then**
    Set $\underline{p}_{t+1}^j = \frac{p_{t+1}^j}{2}, \eta_{t+1,j} = \beta\eta_{t,i}$,
  else
    Set $\underline{p}_{t+1}^j = \underline{p}_t^j, \eta_{t+1,j} = \eta_{t,i}$.
  Set $\rho_{t+1}^j = \frac{1}{\underline{p}_{t+1}^j}$.
Return p_{t+1}, η_{t+1}, $\underline{p}_{t+1}$ and ρ_{t+1}^j.

Corral. We used the **Corral** Algorithm as described in Agarwal et al. (2017) and Pacchiano et al. (2020b). Since we work with stochastic base algorithms we use the Stochastic Corral version of Pacchiano et al. (2020b) where the base algorithms are updated with the observed reward r_t instead of the importance sampling version required by the original **Corral** algorithm of Agarwal et al. (2017). The pseudo-code is in Algorithm 2. In accordance with theoretical results we set $\eta = \Theta(\frac{1}{\sqrt{T}})$. We test the performance of the **Corral** meta-algorithm with different settings of the initial learning rate $\eta \in \{.1/\sqrt{T}, 1/\sqrt{T}, 10/\sqrt{T}\}$. In the table and plots below we call them **CorralLow**, **Corral** and **CorralHigh** respectively. In Table 4 we compare their performance on different experiment benchmarks. We see **Corral** and **CorralHigh** achieve a better formance than **CorralLow**. The performance of **Corral** and **CorralHigh** is similar.

EXP3. At the beginning of each time step the **EXP3** meta-algorithm samples a base learner index $i_t \sim p_t$ from its base learner distribution p_t. The meta-algorithm maintains importance weighted estimator of the cumulative rewards for each base learner R_t^i for all $i \in [M]$. After receiving feedback r_t from base learner i_t the importance weighted estimators are updated as $R_{t+1}^i = R_t^i + \mathbf{1}(i = i_t)\frac{r_t}{p_t^{i_t}}$. The distribution $p_{t+1}^i = (1-\gamma)\exp(\eta R_{t+1}^i)/\sum_{i'}\exp(\eta R_{t+1}^{i'}) + \gamma/M$ where η is a and γ are a learning rate and exploration parameters. In accordance with theoretical results (see for example (Lattimore and Szepesvári, 2020, Th. 11.1)) in our experiments we set the learning rate to $\eta = \sqrt{\frac{\log(M)}{MT}}$ and set the forced exploration parameter $\gamma = \frac{0.1}{\sqrt{T}}$. We

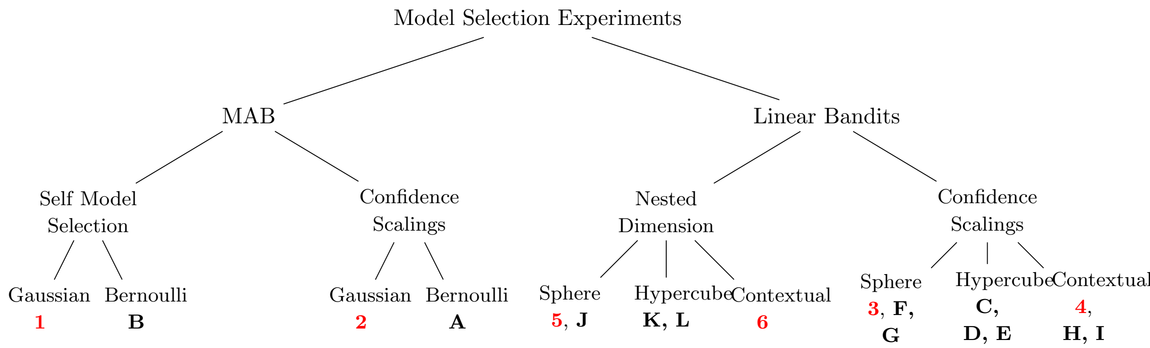


Figure 4: Experiment Map.

test the performance of the **EXP3** meta-algorithm with different settings of the forced exploration parameter $\gamma \in \{0, \frac{.1}{\sqrt{T}}, \frac{1}{\sqrt{T}}\}$. In Table 4 we call them **EXP3Low**, **EXP3** and **EXP3High**. All these different variants have a similar performance.

Greedy. This is a pure exploitation meta-learner. After playing each base learner at least once, the **Greedy** meta-algorithm maintains the same cumulative reward statistics $\{\widehat{u}_t^i\}_{i \in [M]}$ as D^{3}RB and ED^{2}RB. The base learner i_t chosen at time t is $i_t = \text{argmax}_{i \in [M]} \frac{u_t^i}{n_t^i}$.

UCB. We use the same **UCB** algorithm as described in Section 2.2. We set the scaling parameter $c = 1$.

D^{3}RB and ED^{2}RB. These are the algorithms in Algorithm 1. We set therein $c = 1$ and $d_{\min} = 1$.

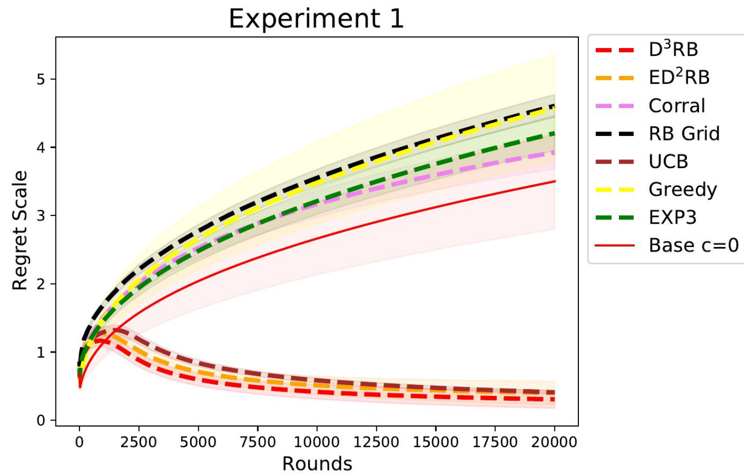
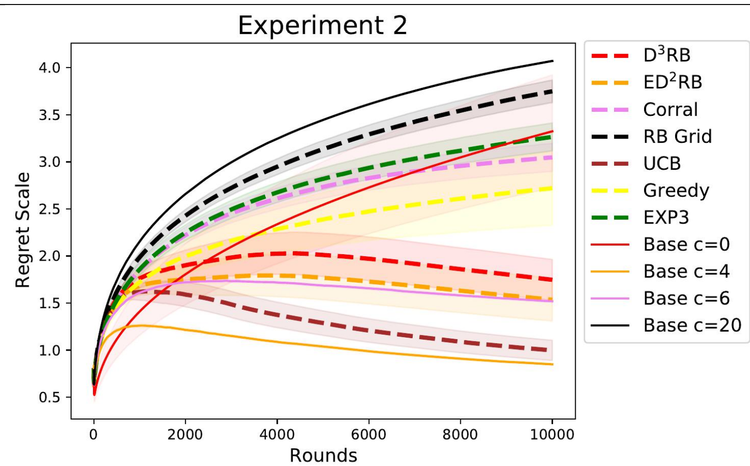
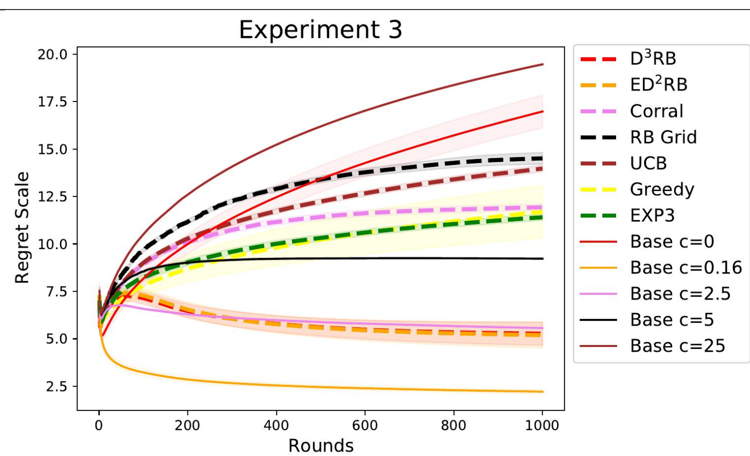
11.2 Base Learners

All base learners have essentially been described, except for the Linear Thompson Sampling Algorithm (LinTS) algorithm, which was used in all our linear experiments.

In our implementation we use the algorithm described as in Abeille and Lazaric (2017). On round t the Linear Thompson Sampling algorithm has played $x_1, \cdots x_{t-1} \subset \mathbb{R}^d$ with observed responses $r_1, \cdots, r_{t-1}$. The rewards are assumed to be of the form $r_\ell = x_\ell^\top \theta_\star + \xi_t$ for an unknown vector $\theta_\star$ and a conditionally zero mean random variable ξ_t. An empirical model of the unknown vector $\theta_\star$ is produced by fitting a ridge regression least squares estimator $\widehat{\theta}_t = \text{argmin}_\theta \lambda \|\theta\|^2 + \sum_{\ell=1}^{t-1} (x_\ell^\top \theta - r_\ell)^2$ for a user specified parameter $\lambda > 0$. This can be written in closed form as $\widehat{\theta}_t = \left(\mathbf{X}^\top \mathbf{X} + \lambda \mathbb{I}\right)^{-1} \mathbf{X}^\top y$ where $\mathbf{X} \in \mathbb{R}^{t-1 \times d}$ matrix where row ℓ equals x_ℓ. At time t a sample model is computed $\widetilde{\theta}_t = \widehat{\theta}_t + c\sqrt{d} \left(\mathbf{X}^\top \mathbf{X} + \lambda \mathbb{I}\right)^{-1/2} \boldsymbol{\eta}_t$ where $\boldsymbol{\eta}_t \sim \mathcal{N}(\mathbf{0}, \mathbb{I})$ and $c > 0$ is a confidence scaling parameter. This is one of the parameters that we vary in our experiments. If the action set at time t equals $\mathcal{A}_t$ (in the contextual setting $\mathcal{A}_t$ changes every time-step while in the fixed action set linear bandits case it) the action $x_t = \text{argmax}_{x \in \mathcal{A}_t} x_t^\top \widetilde{\theta}_t$. In our experiments $\lambda = 1$ and $\theta_\star$ is set to a scaled version of the vector $(0, \cdots, d-1)$. In the detailed experiment description below we specify the precise value of $\theta_\star$ in each experiment.

11.3 Detailed Experiments Description

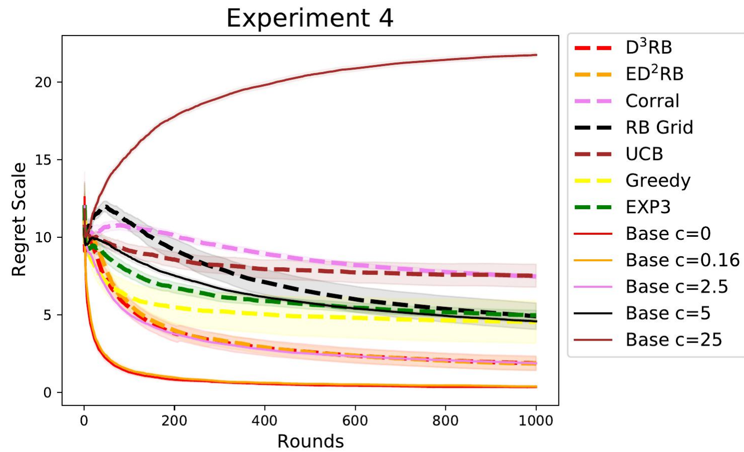
Figure 4 illustrates the overall structure of our experiments. Experiments 1 through 6 are those also reported in the main body of the paper. The below table contains a detailed description of each experiment, together with the associated evidence in the form of learning curves (regret scale vs. rounds). Finally, Table 2 contains the final (average) cumulative regret for each meta-learner on each experiment.

<table>
<tr><th>Description</th><th>Figure</th></tr>
<tr><td>Experiment 1. Self model selection for a 5 armed Gaussian bandit problem with means $[.5, 1, .2, .1, .6]$ and standard deviations equal to 1. The base learners are UCB algorithms with a confidence scaling $c = 0$. This reduces them to instances of Greedy. It is well known that greedy algorithms do not satisfy a sublinear expected regret bound. These results show that combining multiple instances of Greedy via self model selection can turn them into algorithms with a sublinear regret guarantee. We initialize 10 Greedy base learners.</td><td>
</td></tr>
<tr><td>Experiment 2. Model selection for a 5 armed Gaussian bandit problem with means $[.5, 1, .2, .1, .6]$ and standard deviations equal to 1 where we select among 4 UCB base learners with confidence scalings in $\{0, 4, 6, 20\}$.</td><td>
</td></tr>
<tr><td>Experiment 3. Linear bandits model selection where we select among LinTS base learners with different confidence scalings. The action set is the 10 dimensional unit sphere. The $\theta_\star$ vector equals $(0, \ldots, 9)/\|(0, \ldots, 9)\| * 5$. The base learners are LinTS instances with confidence scalings in $\{0, .16, 2.5, 5, 25\}$.</td><td>
</td></tr>
</table>

<table>
<tr><th>Description</th><th>Figure</th></tr>
<tr><td>

Experiment 4. Contextual linear bandits model selection where we select among LinTS base learners with different confidence scalings. The contexts are generated by producing 10 i.i.d. uniformly distributed vectors from the unit sphere. The ambient space dimension is $d = 10$. The $\theta_\star$ vector is $(0, \ldots, 9)$. The base learners are LinTS instances with confidence scalings in $\{0, .16, 2.5, 5, 25\}$.

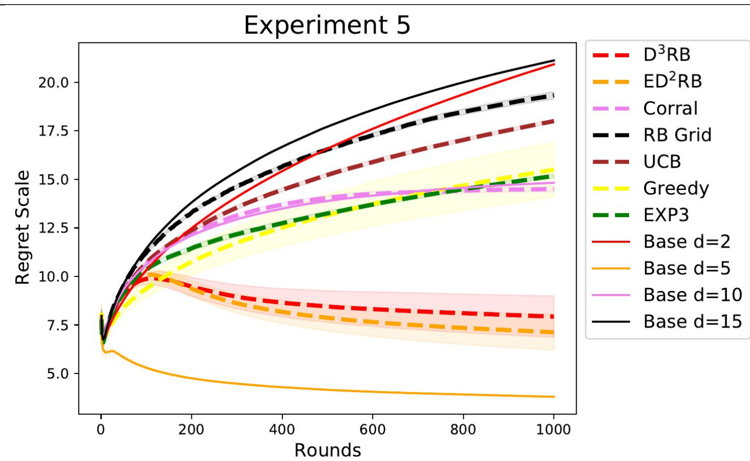
</td><td>



</td></tr>
<tr><td>

Experiment 5. Nested linear bandits model selection where we select among different LinTS base learners with different ambient dimensions. The action set is the unit sphere and the true ambient dimension equals 5. The $\theta_\star$ vector is $(0, 1, 2, 3, 4)$ and the base learners are LinTS instances with dimensions $d = 2, 5, 10, 15$, and confidence scaling 2.

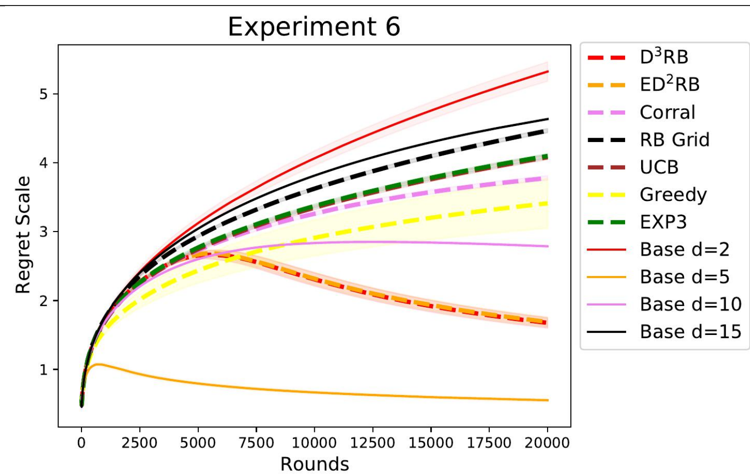
</td><td>



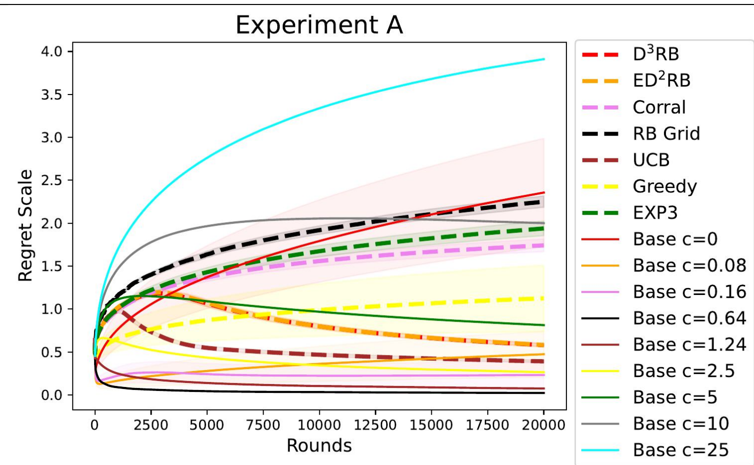
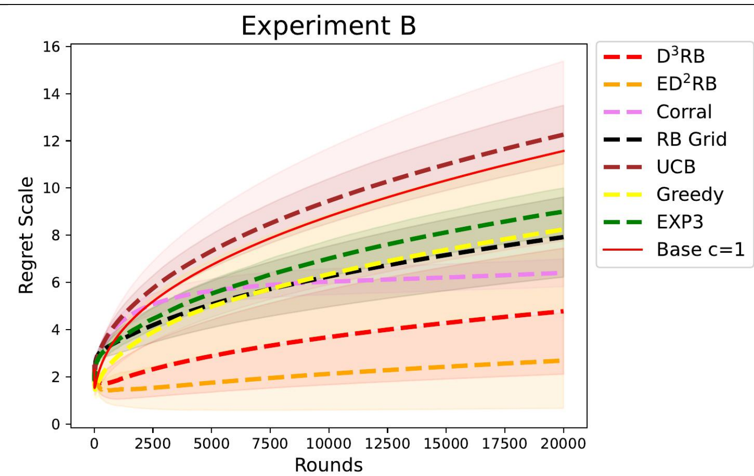
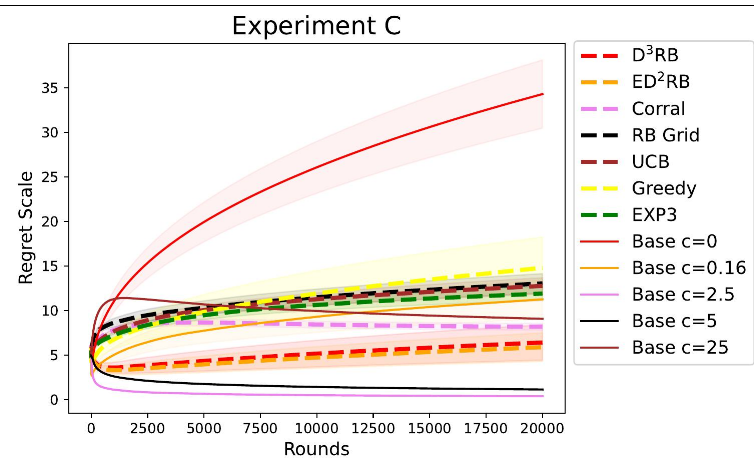
</td></tr>
<tr><td>

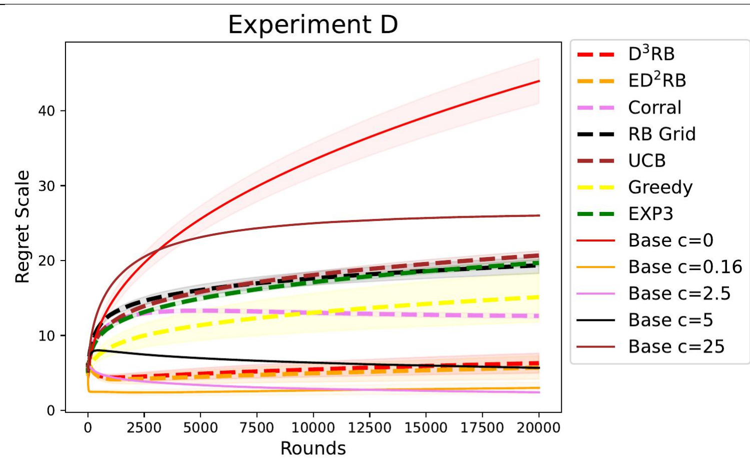
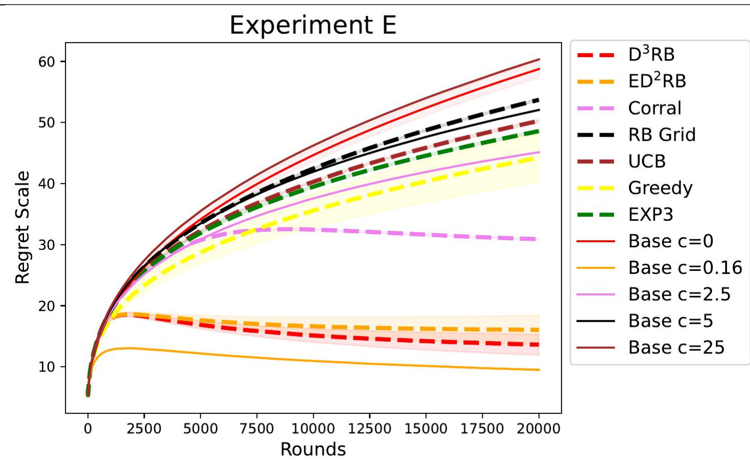
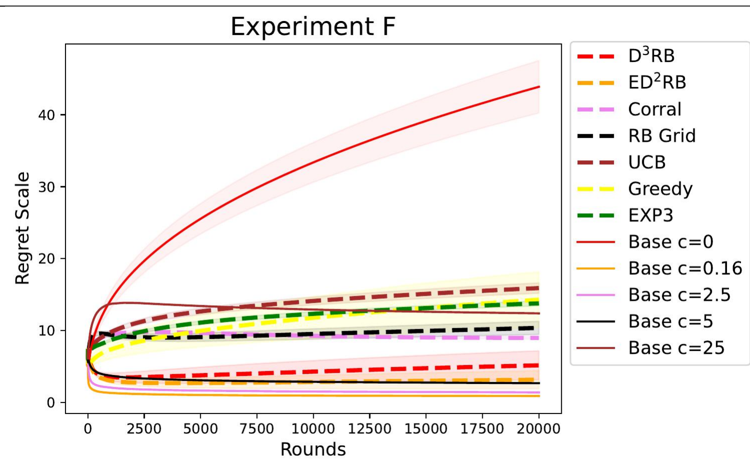
Experiment 6. Nested contextual linear bandits model selection where we select among different LinTS base learners with different ambient dimensions. The context set is generated by sampling 10 i.i.d. vectors from the unit sphere. The true ambient dimension equals 5. The $\theta_\star$ vector equals $(0, 1, 2, 3, 4)/\|(0, 1, 2, 3, 4)\|$ and the base learners are LinTS instances with dimensions $d = 2, 5, 10, 15$ and confidence scaling 2.

</td><td>



</td></tr>
</table>

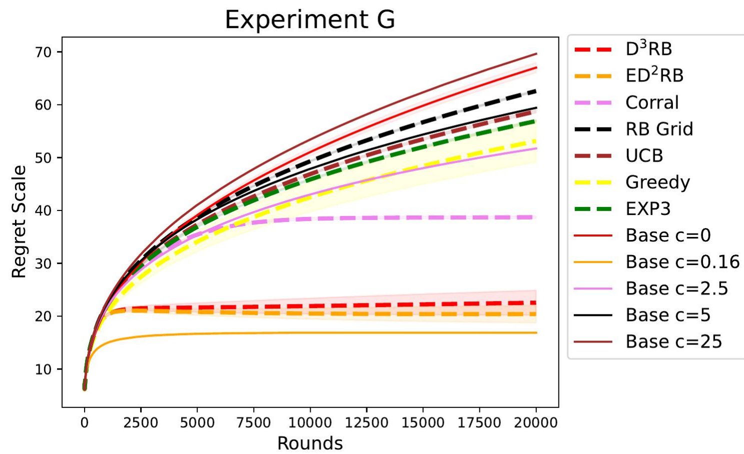
Description	**Figure**
Experiment A. Selecting among different confidence scalings in a 4 armed Bernoulli bandit problem with mean rewards $[.1, .2, .5, .8]$. We select among 9 **UCB** base learners with confidence scalings in $\{0, .08, .16, .64, 1.24, 2.5, 5, 10, 25\}$.	
Experiment B. Selecting among different confidence scalings in a 2 armed bandit problem with reward distributions $r \sim p_1$ and $r \sim p_2$ where samples from p_i are of the form $30b_i$ where b_i are two Bernoulli variables with means in $\{.1, .2\}$. We test self-model selection among 10 **UCB** base learners with confidence scalings 1.	
Experiment C. Linear bandit model selection where we select among LinTS base learners with different confidence scalings. The action set equals the $d = 5$ dimensional hypercube (i.e., the arm set equals $\{-1, 1\}^d$). The $\theta_\star$ vector equals $(0, \dots, 4)/\|(0, \dots, 4)\| * 5$. The base learners are LinTS instances with confidence scalings in $\{0, .16, 2.5, 5, 25\}$.	

Description	Figure
Experiment D. Linear bandit model selection where we select among LinTS base learners with different confidence scalings. The action set equals the $d = 10$ dimensional hypercube (i.e., the arm set equals $\{-1/\sqrt{d}, 1/\sqrt{d}\}^d$). The $\theta_\star$ vector equals $(0, \ldots, 9)/\|(0, \ldots, 9)\| * 5$. The base learners are LinTS instances with confidence scalings in $\{0, .16, 2.5, 5, 25\}$.	
Experiment E. Linear bandit model selection where we select among LinTS base learners with different confidence scalings. The action set equals the $d = 100$ dimensional hypercube (i.e., the arm set equals $\{-1/\sqrt{d}, 1/\sqrt{d}\}^d$). The $\theta_\star$ vector equals $(0, \ldots, 99)/\|(0, \ldots, 99)\| * 5$. The base learners are LinTS instances with confidence scalings in $\{0, .16, 2.5, 5, 25\}$.	
Experiment F. Linear bandit model selection where we select among LinTS base learners with different confidence scalings. The action set equals the $d = 5$ dimensional unit sphere. The $\theta_\star$ vector equals $(0, \ldots, 4)/\|(0, \ldots, 4)\| * 5$. The base learners are LinTS instances with confidence scalings in $\{0, .16, 2.5, 5, 25\}$.	

<table>
<tr><th>Description</th><th>Figure</th></tr>
<tr><td>

Experiment G. Linear bandit model selection where we select among LinTS base learners with different confidence scalings. The action set equals the $d = 100$ dimensional unit sphere. The $\theta_\star$ vector equals $(0, \ldots, 99)/\|(0, \ldots, 99)\| * 5$. The base learners are LinTS instances with confidence scalings in $\{0, .16, 2.5, 5, 25\}$.

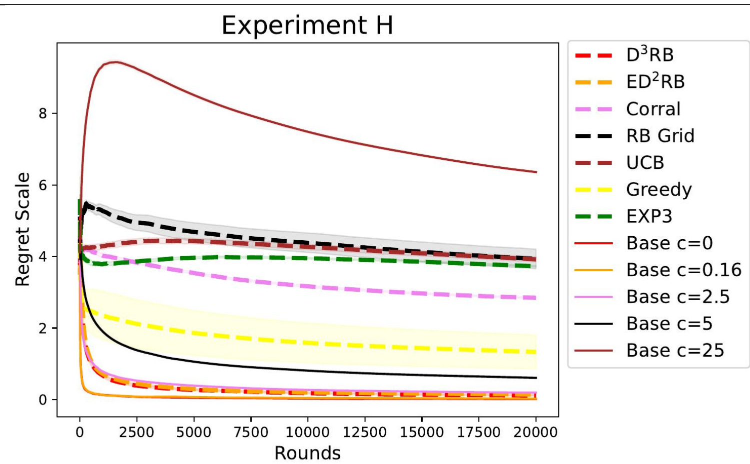
</td><td>



</td></tr>
<tr><td>

Experiment H. Contextual linear bandit model selection where we select among LinTS base learners with different confidence scalings. The contexts are generated by producing 10 i.i.d. uniformly distributed vectors from the unit sphere. The ambient space dimension equals $d = 5$. The $\theta_\star$ vector equals $(0, \ldots, 4)/\|(0, \ldots, 4)\| * 5$. The base learners are LinTS instances with confidence scalings in $\{0, .16, 2.5, 5, 25\}$.

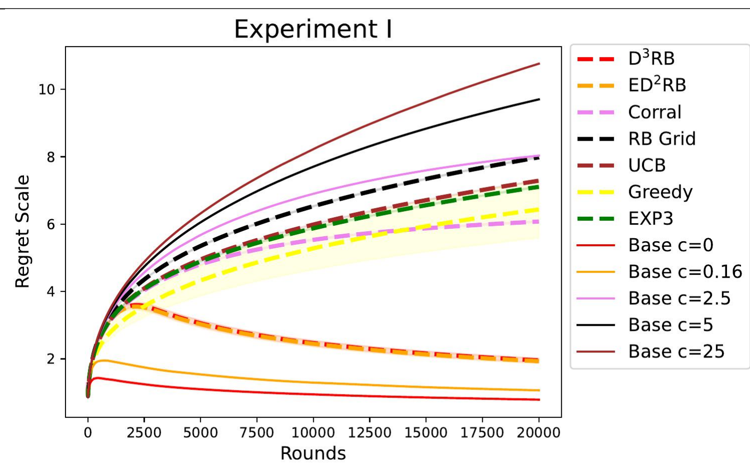
</td><td>



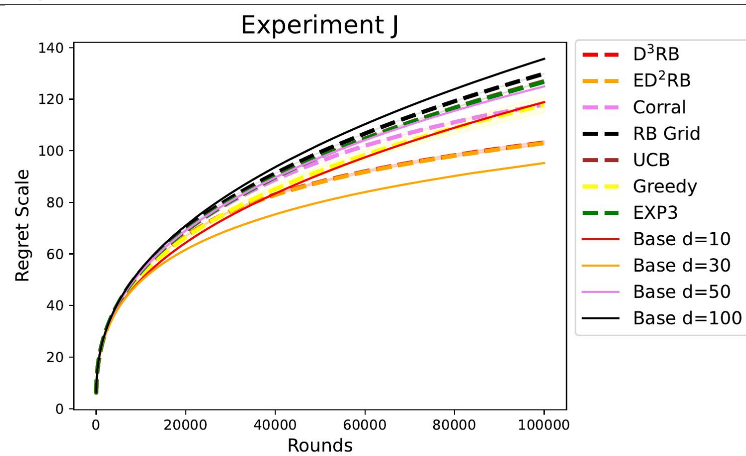
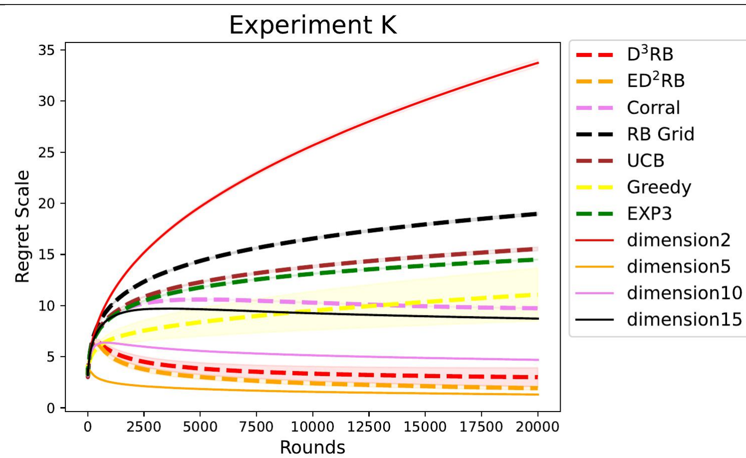
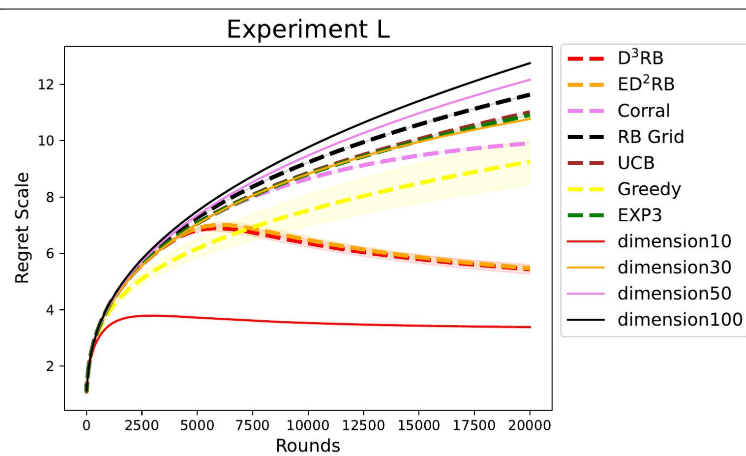
</td></tr>
<tr><td>

Experiment I. Contextual linear bandit model selection where we select among LinTS base learners with different confidence scalings. The contexts are generated by producing 10 i.i.d. uniformly distributed vectors from the unit sphere. The ambient space dimension equals $d = 100$. The $\theta_\star$ vector is $(0, \ldots, 99)/\|(0, \ldots, 99)\| * 5$. The base learners are LinTS instances with confidence scalings in $\{0, .16, 2.5, 5, 25\}$.

</td><td>



</td></tr>
</table>

Description	**Figure**
Experiment J. Nested linear bandit model selection where we select among different LinTS base learners with different ambient dimensions. The action set is the unit sphere and the true ambient dimension is 30. The $\theta_\star$ vector is $(0,1,\ldots,29)/\|(0,1,\ldots,29)\| * 5$ and the base learners are LinTS instances with dimensions $d = 10, 30, 50, 100$, and confidence scaling 2.	
Experiment K. Nested linear bandit model selection where we select among different LinTS base learners with different ambient dimensions. The action set is the hypercube and the true ambient dimension is 5. The $\theta_\star$ vector is $(0,1,2,3,4)$ and the base learners are LinTS instances with dimensions $d = 2, 5, 10, 15$ and confidence scaling 2.	
Experiment L. Nested linear bandit model selection where we select among different LinTS base learners with different ambient dimensions. The action set is the hypercube and the true ambient dimension is 5. The $\theta_\star$ vector is $(0,1,\ldots,29)/\|(0,1,\ldots,29)\| * 5$ and the base learners are LinTS instances with dimensions $d = 10, 30, 50, 100$ and confidence scaling 2.	

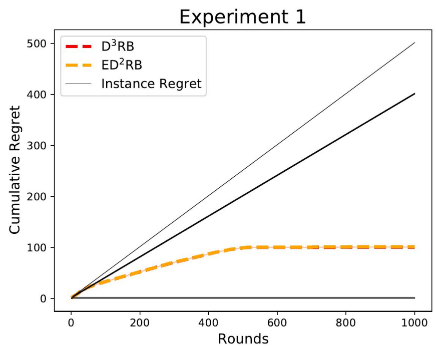


Figure 5: Experiment 1 (see Table 1 for reference). Regret for **D^{3}RB** and **ED^{2}RB** (Algorithm 1) on a single realization.

Table 2: Comparison of meta learners over the various experiments (Experiment 1 through 6, and Experiment A through L). The lines corresponding to Experiments ("Nm" = Name) 1–6 are also contained in the main body of the paper. We report the cumulative regret (averaged over 100 repetitions $\pm 2\times$standard error) at the end of the sequence of rounds. In bold is the best performer for each environment. In order to save horizontal space, we abbreviated the arm names as follows: "Gauss" is Gaussian, "Bern" is Bernoulli, "Sph" is Sphere, "Hyper" is Hypercube, and "Cont" is Contextual. Moreover, the column "Lrn." indicates the kind of base learner used in the corresponding experiment.

Nm	Env.	Lrn.	Task	Arms	D^{3}RB	ED^{2}RB	Corral	RB Grid	UCB	Greedy	EXP3
1.	MAB	UCB	self	Gauss	**431 ± 182**	560 ± 240	5498 ± 340	6452 ± 230	574 ± 34	6404 ± 1102	5892 ± 356
B.	MAB	UCB	self	Bern	6694 ± 3738	**3765 ± 2834**	8972 ± 818	11093 ± 2375	17174 ± 1737	11546 ± 4698	12606 ± 1399
2.	MAB	UCB	conf	Gauss	1608 ± 198	1413 ± 208	2807 ± 138	3452 ± 110	**918 ± 98**	2505 ± 362	3007 ± 136
A.	MAB	UCB	conf	Bern	811 ± 27	813 ± 28	2439 ± 102	3152 ± 91	**547 ± 20**	1576 ± 541	2715 ± 118
3.	LB	LinTS	conf	Sph	1150 ± 134	**1135 ± 148**	2605 ± 38	3169 ± 66	3052 ± 36	2553 ± 302	2491 ± 36
F.	LB	LinTS	conf	Sph	7251 ± 2820	**4458 ± 1748**	12594 ± 237	14536 ± 1278	22286 ± 1000	20034 ± 5416	19316 ± 488
G.	LB	LinTS	conf	Sph	31585 ± 3269	**28579 ± 2301**	54230 ± 318	87652 ± 224	82259 ± 344	74433 ± 5596	79706 ± 277
C.	LB	LinTS	conf	Hyper	8971 ± 2783	**8237 ± 2246**	11453 ± 538	18264 ± 1528	17868 ± 1277	20701 ± 4835	16663 ± 1091
D.	LB	LinTS	conf	Hyper	8831 ± 1855	**8006 ± 2174**	17655 ± 451	27119 ± 1526	28957 ± 853	21167 ± 4686	27576 ± 582
E.	LB	LinTS	conf	Hyper	**19088 ± 2414**	22474 ± 3343	43281 ± 191	75233 ± 255	70389 ± 318	61976 ± 5567	68045 ± 306
4.	CLB	LinTS	conf	Cont	411 ± 100	**406 ± 94**	1632 ± 30	1073 ± 184	1644 ± 160	991 ± 298	1086 ± 70
H.	CLB	LinTS	conf	Cont	**165 ± 27**	199 ± 24	3986 ± 31	5506 ± 384	5490 ± 100	1867 ± 666	5220 ± 31
I.	CLB	LinTS	conf	Cont	2735 ± 95	**2705 ± 101**	8510 ± 35	11184 ± 26	10210 ± 25	9012 ± 1184	9954 ± 30
5.	LB	LinTS	dim	Sph	1733 ± 230	**1556 ± 198**	3166 ± 26	4223 ± 40	3932 ± 16	3385 ± 306	3315 ± 20
J.	LB	LinTS	dim	Sph	**93871 ± 107**	93914 ± 119	96194 ± 39	97828 ± 25	96703 ± 31	93681 ± 926	96659 ± 27
K.	LB	LinTS	dim	Hyper	4188 ± 1288	**2681 ± 250**	13634 ± 57	26570 ± 174	21765 ± 255	15505 ± 3622	20308 ± 69
L.	LB	LinTS	dim	Hyper	**7614 ± 233**	7660 ± 126	13874 ± 38	16287 ± 19	15410 ± 20	12953 ± 1132	15275 ± 22
6.	CLB	LinTS	dim	Cont	**2347 ± 102**	2365 ± 96	5294 ± 44	6258 ± 38	5718 ± 50	4778 ± 506	5742 ± 46

Table 3: Comparison of the **CorralLow**, **Corral** and **CorralHigh** meta learners over the various experiments (Experiment 1 through 6, and Experiment A through L). All of the experiments of this table were run for 20000 time-steps. This is in contrast with the results presented for Experiments 1-6 in Table 1 and in Table 2. We report the cumulative regret (averaged over 100 repetitions $\pm 2\times$standard error) at the end of the sequence of rounds. In bold is the best performer for each environment.

Name	Env.	Learners	Task	Arms	CorralLow	Corral	CorralHigh
1.	MAB	UCB	self	Gaussian	6609 ± 433	5498 ± 340	$\mathbf{2598 \pm 256}$
B.	MAB	UCB	self	Bernoulli	15324 ± 1502	$\mathbf{8972 \pm 818}$	10107 ± 1516
2.	MAB	UCB	conf	Gaussian	5069 ± 275	4670 ± 251	$\mathbf{3093 \pm 175}$
A.	MAB	UCB	conf	Bernoulli	2742 ± 113	2439 ± 102	$\mathbf{319 \pm 9}$
3.	LB	LinTS	conf	Sphere	32683 ± 559	$\mathbf{19249 \pm 212}$	24287 ± 850
F.	LB	LinTS	conf	Sphere	20073 ± 752	$\mathbf{12594 \pm 237}$	15790 ± 629
G.	LB	LinTS	conf	Sphere	80230 ± 272	54230 ± 318	$\mathbf{51499 \pm 1535}$
C.	LB	LinTS	conf	Hypercube	18783 ± 911	11453 ± 538	$\mathbf{4369 \pm 198}$
D.	LB	LinTS	conf	Hypercube	27575 ± 730	$\mathbf{17655 \pm 451}$	20190 ± 879
E.	LB	LinTS	conf	Hypercube	68764 ± 349	43281 ± 191	$\mathbf{41918 \pm 1236}$
4.	CLB	LinTS	conf	Context.	11880 ± 97	$\mathbf{7606 \pm 93}$	10646 ± 444
H.	CLB	LinTS	conf	Context.	5269 ± 31	$\mathbf{3986 \pm 31}$	4624 ± 160
I.	CLB	LinTS	conf	Context.	10045 ± 26	$\mathbf{8510 \pm 35}$	6823 ± 171
5.	LB	LinTS	dim	Sphere	44957 ± 73	$\mathbf{22614 \pm 69}$	36384 ± 926
J.	LB	LinTS	dim	Sphere	96655 ± 34	96194 ± 39	$\mathbf{95394 \pm 153}$
K.	LB	LinTS	dim	Hypercube	20533 ± 86	$\mathbf{13634 \pm 57}$	14237 ± 335
L.	LB	LinTS	dim	Hypercube	15312 ± 23	13874 ± 38	$\mathbf{12295 \pm 259}$
6.	CLB	LinTS	dim	Context.	$\mathbf{1295 \pm 50}$	5294 ± 44	4711 ± 101

Table 4: Comparison of the **EXP3Low**, **EXP3** and **EXP3High** meta learners over the various experiments (Experiment 1 through 6, and Experiment A through L). All of the experiments of this table were run for 20000 time-steps. This is in contrast with the results presented for Experiments 1-6 in Table 1 and in Table 2. We report the cumulative regret (averaged over 100 repetitions $\pm 2\times$standard error) at the end of the sequence of rounds. In bold is the best performer for each environment. All these algorithms have similar performance.

Name	Env.	Learners	Task	Arms	EXP3Low	EXP3	EXP3High
1.	MAB	UCB	self	Gaussian	$\mathbf{5733 \pm 376}$	5892 ± 356	5743 ± 354
B.	MAB	UCB	self	Bernoulli	13764 ± 1498	$\mathbf{12606 \pm 1399}$	13308 ± 1337
2.	MAB	UCB	conf	Gaussian	5224 ± 299	5332 ± 269	$\mathbf{5136 \pm 279}$
A.	MAB	UCB	conf	Bernoulli	$\mathbf{2607 \pm 109}$	2715 ± 118	2613 ± 113
3.	LB	LinTS	conf	Sphere	31546 ± 356	$\mathbf{31252 \pm 463}$	31256 ± 441
F.	LB	LinTS	conf	Sphere	19453 ± 535	$\mathbf{19316 \pm 488}$	19844 ± 529
G.	LB	LinTS	conf	Sphere	79679 ± 276	79706 ± 277	$\mathbf{79311 \pm 290}$
C.	LB	LinTS	conf	Hypercube	17264 ± 938	$\mathbf{16663 \pm 1091}$	17842 ± 956
D.	LB	LinTS	conf	Hypercube	$\mathbf{26789 \pm 570}$	27576 ± 582	27355 ± 599
E.	LB	LinTS	conf	Hypercube	$\mathbf{67479 \pm 325}$	68045 ± 306	67756 ± 270
4.	CLB	LinTS	conf	Context.	$\mathbf{11877 \pm 207}$	11885 ± 91	11921 ± 71
H.	CLB	LinTS	conf	Context.	5293 ± 34	$\mathbf{5220 \pm 31}$	5254 ± 35
I.	CLB	LinTS	conf	Context.	$\mathbf{9934 \pm 31}$	9954 ± 30	9959 ± 27
5.	LB	LinTS	dim	Sphere	46136 ± 55	46129 ± 52	$\mathbf{46127 \pm 54}$
J.	LB	LinTS	dim	Sphere	96663 ± 26	96659 ± 27	$\mathbf{96643 \pm 27}$
K.	LB	LinTS	dim	Hypercube	$\mathbf{20294 \pm 60}$	20308 ± 69	20315 ± 86
L.	LB	LinTS	dim	Hypercube	15280 ± 23	15275 ± 22	$\mathbf{15256 \pm 22}$
6.	CLB	LinTS	dim	Context.	5761 ± 59	5742 ± 46	$\mathbf{5693 \pm 49}$