Bayesian decision-making under misspecified priors with applications to meta-learning

Anonymous Author(s)
Affiliation
Address
email

Abstract

Thompson sampling and other Bayesian sequential decision-making algorithms are among the most popular approaches to tackle explore/exploit trade-offs in (contextual) bandits. The choice of prior in these algorithms offers flexibility to encode domain knowledge but can also lead to poor performance when misspecified. In this paper, we demonstrate that performance degrades gracefully with misspecification. We prove that the expected reward accrued by Thompson sampling (TS) with a misspecified prior differs by at most $\tilde{O}(H^2\epsilon)$ from TS with a well specified prior, where ϵ is the total-variation distance between priors and H is the learning horizon. Our bound does not require the prior to have any parametric form. For priors with bounded support, our bound is independent of the cardinality or structure of the action space, and we show that it is tight up to universal constants in the worst case. Building on our sensitivity analysis, we establish generic PAC guarantees for algorithms in the recently studied Bayesian meta-learning setting and derive corollaries for various families of priors. Our results generalize along two axes: (1) they apply to a broader family of Bayesian decision-making algorithms, including a Monte-Carlo implementation of the knowledge gradient algorithm (KG), and (2) they apply to Bayesian POMDPs, the most general Bayesian decision-making setting, encompassing contextual bandits as a special case. Through numerical simulations, we illustrate how prior misspecification and the deployment of one-step look-ahead (as in KG) can impact the convergence of meta-learning in multi-armed and contextual bandits with structured and correlated priors.

1 Introduction

Bayesian decision-making algorithms are widely popular, due to both strong empirical performance and the flexibility afforded by incorporating inductive biases through priors. However, in practical applications, any chosen prior is at best an approximation of the true environment in which the algorithm is deployed. This raises a critical question:

How sensitive are Bayesian decision-makers to prior misspecification?

For decision-making problems with a very large horizon, it suffices that the misspecified prior places a vanishingly small probability mass on the ground truth environment; this condition is referred to informally as a "grain of truth." This is because, in the large-horizon limit, Bayesian algorithms (like many non-Bayesian methods) should converge to the optimal policy.

But in many practical settings, decision-making takes place on shorter time scales. Consider a news recommendation website that, when presented with a new user, sequentially offers a selection of currently trending articles. Such a system may only have a few opportunities to make recommendations before the user decides to navigate away, leaving little time to correct for misspecified or

Submitted to 35th Conference on Neural Information Processing Systems (NeurIPS 2021). Do not distribute.

underspecified prior knowledge. Such examples are described more broadly by the meta-learning paradigm, where a single learning agent must complete multiple disparate-though-related tasks.

In meta-learning problems, and in short time horizon problems more broadly, the "grain of truth" argument paints a rather uninformative picture. Consequently, recent work has begun to explore sensitivity bounds in shorter horizon applications [LL16, KKZ+21]. However, these recent works focus on particular classes of priors and/or reward models, as well as on the Thompson sampling algorithm specifically. Notably, this leaves open questions about the extent to which prior sensitivity is determined by properties of the Bayesian decision-making algorithm, of the rewards model, and of the prior itself.

1.1 Our Contributions

Motivated by meta-learning problems with short task horizons, we establish general, distribution-independent, and worst-case optimal bounds on the sensitivity of Bayesian algorithms to prior misspecification. We focus on the Bayesian bandit setting, where a mean-vector "environment" $\boldsymbol{\mu}$ is drawn from a distribution P, and rewards for each action are drawn in accordance with $\boldsymbol{\mu}$. We study the performance of Bayesian algorithms which operate according to a *misspecified* prior P'.

Sensitivity of Thompson Sampling and Related Bayesian Bandit Algorithms. As a concrete example, we consider the expected reward obtained by Thompson sampling with misspecified prior P' under environments drawn from true prior P.

When the mean rewards lie in the range $[0, 1]$, as in the Bernoulli reward setting, we show that that the difference in expected reward between Thompson sampling with P' and with P is at most twice the total variation distance between P and P' multiplied by the *square* of the horizon length. We prove a lower bound demonstrating that, for worst-case priors, this result is tight up to constants. Moreover, our upper bound holds for any two priors P and P', and suffers no dependence on the complexity of the decision space.

We extend this result in two directions. First, we remove the boundedness requirement on the mean reward range, showing that so long as certain tail probability conditions on the prior means are satisfied, a similar result holds. Second, we generalize beyond Thompson sampling, bounding the prior sensitivity of a broad class of Bayesian bandit algorithms, which we term *n-Monte Carlo algorithms*. Our lower bounds extend to this class, verifying sharp dependence on the parameter n.

Sample Complexity of Bayesian Meta-Learning. We apply our prior sensitivity results to the Bayesian bandit meta-learning setting, in which a meta-learner iteratively interacts on bandit instances that are sampled from an unknown prior distribution. Motivated by our sensitivity analysis we describe a generic algorithmic recipe for Bayesian meta-learning, in which the meta-learner explores for several episodes to estimate the prior and then exploits by instantiating a Bayesian decision-maker with the learner prior. We formally consider two instantiations of this setup: (1) the Beta-Bernoulli setting where the rewards are Bernoulli and the prior is a product of Beta distributions and (2) the Gaussian-Gaussian setting where the rewards are Gaussian and the prior is a Gaussian (with arbitrary covariance structure) over the means. We note that the Gaussian-Gaussian setting was recently studied in [KKZ+21] but only for the diagonal covariance setting.

Bayesian Decision-Making Beyond the Bandit Setting. A striking feature of our proof is that it makes no explicit reference to the structure of bandit decision-making. As a consequence, our results extend naturally to both contextual bandits and the most general Bayesian decision-making problem: Bayesian POMDPs. While our sensitivity bounds hold almost verbatim in these settings, we note that estimating the prior may be statistically much more challenging in these scenarios, so there is no free lunch. To facilitate readability of the paper, we defer all further discussion and formal results to Appendix E.

Experimental results. We complement our meta-learning theory with synthetic experiments in multi-armed bandit and contextual bandit settings. Our experiments show the benefits of (a) meta-learning broadly, (b) estimating higher-order moments of the prior distribution, and (c) using less myopic algorithms like the Knowledge Gradient [RPF12] over Thompson sampling when faced with structured environments.

1.2 Related Work

Bayesian Decision-Making. Bayesian decision-making broadly refers to a class of algorithms that use Bayesian methods to estimate various problem parameters, and then derive decision/allocation rules from these estimates. The study of Bayesian decision-making began with the seminal work of Thompson [Tho33], who introduced the Thompson sampling algorithm for adaptive experiment design in clinical trials. Thompson sampling later gained popularity in the reinforcement learning community as a means to solve multi-arm bandit and tabular reinforcement learning problems [Str00, OVR17], and has been extended in many directions [AL17, AL18, GMM14]. Recent years have seen the proliferation of other Bayesian decision-making and learning algorithms, including Information Directed Sampling [RVR16], Top-Two Thompson Sampling [Rus16], and Knowledge-Gradient [RPF12].

Sensitivity Analysis and Frequentist Regret. The field of robust Bayesian analysis examines the sensitivity of Bayesian inference to prior and model misspecification [c.f., BMP$^+$94]. These approaches typically do not consider decision-making, so they do not account for multi-step adaptive sampling inherent in our setting. More recent works study frequentist regret for Thompson sampling [AG12, KKM12]. These guarantees can be interpreted as controlling the sensitivity to arbitrary degrees of prior misspecification, but consequently, they do not provide a precise picture of how misspecification affects performance. Moreover, frequentist guarantees for Thompson sampling focus on long learning horizons, so they are less relevant in the context of meta-learning with many short-horizon tasks.

Short-Horizon Sensitivity. Most closely related to our paper are two previous works on sensitivity of Thompson sampling to small amounts of misspecification in short horizon settings. [LL16] study the sensitivity of Thompson sampling for two-arm bandits when the prior has finite support. More recently, [KKZ$^+$21] study meta-learning with Thompson sampling and derive sensitivity bounds for Thompson sampling in multi-arm bandits with Gaussian rewards and independent-across-arm Gaussian priors. In contrast to both of these works, the bounds presented in this work apply to arbitrary families of priors, more general decision-making problems, and to more general families of decision-making algorithms. Further, as illustrated in Remark 1, our bounds are also tighter than those achieved by [KKZ$^+$21] when specialized to their precise setting. Finally, our lower bounds demonstrate that the square-horizon factor incurred in [KKZ$^+$21] is unavoidable for worst-case priors (though perhaps not for their special case).

Meta-learning and Meta-RL. Meta-learning is a classical learning paradigm in which a learner faces many distinct-but-related tasks [Thr96, Thr98, Bax98, Bax00, HYC01]. While the classical work primarily considered supervised learning tasks, recent, predominantly empirical, work has focused on meta-reinforcement learning (Meta-RL), where each task is itself a decision-making problem [c.f., WKNT$^+$16, DSC$^+$16]. This includes some Bayesian approaches [HGH$^+$19]. While there have been some theoretical results on Meta-RL in multi-arm and contextual bandit settings [ALB13, CLP20], apart from [KKZ$^+$21] we are not aware of other theoretical treatments with a Bayesian flavor.

2 Setting and Notation

Throughout, we use bold $\mathbf{v}$ to denote vectors and non-bold v_a to denote scalar indices. When the vector $\mathbf{v}_h$ has a subscript, $v_{h,a}$ denotes its coordinates.

Bayesian Bandit Learning under Misspecification. A Bayesian bandit learning instance is specified by (a) an abstract action space $\mathcal{A}$, (b) a parametric family of priors P_θ indexed by parameters $\theta \in \Theta$ over mean vectors $\boldsymbol{\mu} \in \mathbb{R}^{\mathcal{A}}$ with coordinates μ_a, and (c) a function $\mathcal{D}(\cdot) : \mathbb{R}^{\mathcal{A}} \to \Delta(\mathbb{R}^{\mathcal{A}})$ mapping mean vectors $\boldsymbol{\mu}$ to reward vectors $\mathbf{r} \in \mathbb{R}^{\mathcal{A}}$ such that the mean reward under $\mathcal{D}(\boldsymbol{\mu})$ is $\boldsymbol{\mu}$: $\mathbb{E}_{\mathbf{r}\sim\mathcal{D}(\boldsymbol{\mu})}[\mathbf{r}] = \boldsymbol{\mu}$.[1] Note that this general setup allows the prior P_θ to encode complex dependencies between the mean rewards μ_a of actions $a \in \mathcal{A}$.

We consider an episodic bandit protocol with horizon H. First, $\boldsymbol{\mu} \sim P_\theta$ is drawn from the prior. Then, at each time step $h = 1, 2, \ldots, H$, the learner's policy, specified by an algorithm alg, selects an action $a_h \in \mathcal{A}$. Simultaneously, a reward vector $\mathbf{r}_h$ is drawn independently from $\mathcal{D}(\boldsymbol{\mu})$, and the

[1] In fact, our analysis extends to more general cases where the reward distribution is parameterized by more than just the mean vectors, but we restrict ourselves to the current setting for ease of exposition.

learner observes reward $r_h = r_{h,a_h}$. The choice of action a_h may depend on the partial trajectory $\tau_{h-1} = (a_1, r_1, \dots, a_{h-1}, r_{h-1})$. We let $P_{\theta,\mathsf{alg}}$ denote the joint law over $\boldsymbol{\mu}$, and the full trajectory τ_H, while expectations are denoted $E_{\theta,\mathsf{alg}}$. We abbreviate the full trajectory $\tau = \tau_H$. We denote the cumulative reward

$$R(\theta, \mathsf{alg}) := E_{\theta,\mathsf{alg}}\left[\sum_{h=1}^{H} r_h\right] = E_{\theta,\mathsf{alg}}\left[\sum_{h=1}^{H} \mu_{a_h}\right].$$

Bayesian Learning Algorithms. We study a class of algorithms $\mathsf{alg}(\theta)$ also parameterized by $\theta \in \Theta$. For concreteness, the reader may think of $\mathsf{alg}(\theta)$ as corresponding to Thompson sampling, where the learner internally computes posteriors using P_θ as its prior. More general classes of Bayesian algorithms are defined in Section 3.1. We are interested in the consequences of misspecification; that is, interacting with $\boldsymbol{\mu} \sim P_\theta$, but executing $\mathsf{alg}(\theta')$ for some other $\theta' \neq \theta$. Note that our notation for the induced law on the trajectory is $P_{\theta,\mathsf{alg}(\theta')}$.

Episodic Bayesian Meta-Learning. We apply the above framework to the problem of Bayesian meta-learning. Let $\theta^\star \in \Theta$ be a ground-truth parameter. At each episode $t = 1, 2, \dots, T$, a mean parameter $\boldsymbol{\mu}^{(t)}$ is drawn i.i.d. from $P_{\theta^\star}$. Simultaneously, the learner commits to a (potentially non-Bayesian) exploration strategy $\mathsf{explore}^{(t)}$ and collects the induced trajectory $\tau^{(t)}$. At the end of T episodes, the learner selects a parameter $\hat{\theta} \in \Theta$ as a function of $\tau^{(1)}, \dots, \tau^{(T)}$. The learner's performance is evaluated on the expected reward of the plug-in algorithm on $\hat{\theta}$: $R(\theta^\star, \mathsf{alg}(\hat{\theta}))$.

Further notation. Given two probability distributions P and Q over the same probability space $(\Omega, \mathcal{F})$, we denote their total variation $\mathrm{TV}(P \parallel Q) := \sup_{\mathcal{E}\in\mathcal{F}} |P[\mathcal{E}] - Q[\mathcal{E}]|$ and Kullback-Leibler divergence $\mathrm{KL}(P \parallel Q)$. If P is a joint distribution of random variables $(X, Y, Z, \dots)$, $P(X)$ denotes the marginal of X under P, and $P(Y|X)$ the conditional distribution (as a function of random variable X). We define the diameter of a mean vector as $\mathrm{diam}(\boldsymbol{\mu}) := \sup_{a\in\mathcal{A}} \mu_a - \inf_{a\in\mathcal{A}} \mu_a$, which is a random variable when $\boldsymbol{\mu}$ is drawn from P_θ. Throughout, $\log(\cdot)$ denotes the natural logarithm. Given a space $\mathcal{X}$, we let $\Delta(\mathcal{X})$ be the set of probability distributions on $\mathcal{X}$; see Appendix B.1 for measure-theoretic considerations.

3 Prior Sensitivity in Bayesian Learning

This section states sensitivity bounds for various Bayesian bandit algorithms and families of priors, starting with the concrete instance of Thompson sampling under priors with bounded-range means. Our results extend almost verbatim to more general decision-making tasks such as contextual bandits; see Appendix E for further details. Throughout, we use the fact that the posterior distribution of the mean $\boldsymbol{\mu}$ given trajectories τ_h does not depend on the choice of learning algorithm alg; hence, we denote these posteriors $P_\theta[\cdot \mid \tau_h]$.[2]

Recall the classical Thompson sampling algorithm: at each step h, $\mathsf{TS}(\theta)$ draws a mean $\tilde{\boldsymbol{\mu}}_h \sim P_\theta[\cdot \mid \tau_h]$ and selects the reward-maximizing action $a_h \in \arg\max_a \tilde{\mu}_{h,a}$. We say that the prior P_θ is *B-bounded* if $P_\theta[\mathrm{diam}(\boldsymbol{\mu}) \leq B] = 1$. For Thompson sampling under B-bounded priors, we have the following result:

Corollary 3.1. *Let P_θ be B-bounded. Then, the suboptimality of misspecified Thompson sampling $\mathsf{TS}(\theta')$ on instance θ is at most*

$$|R(\theta, \mathsf{TS}(\theta)) - R(\theta, \mathsf{TS}(\theta'))| \leq 2H^2 \cdot \mathrm{TV}(P_\theta \parallel P_{\theta'}) \cdot B.$$

Corollary 3.1 follows directly from Theorem 3.2, which we state in Section 3.2, and which generalizes the statement of the corollary along two axes: to a more general family of Bayesian algorithms we call "n-Monte Carlo", and to less restrictive conditions on the behavior of $\mathrm{diam}(\boldsymbol{\mu})$, such as sub-Gaussian tails. Due to lack of space, we focus on the first such generalization; the second direction is more technical in nature, and we leave its exposition to Appendix B.2.

[2] Note that these posteriors may be defined even if τ_h is generated by interacting with mean $\boldsymbol{\mu} \sim P_{\theta'}$ drawn from a prior $\theta' \neq \theta$.

3.1 n-Monte Carlo algorithms

Unfortunately, for arbitrary Bayesian bandit algorithms, the behavior under two different priors cannot always be controlled in terms of the total variation distance of their priors. Indeed, consider an algorithm that always pulls a particular arm $a^\star$ if the prior places any probability mass on a mean for which $a^\star$ is best; clearly, this algorithm's behavior is not robust to small changes in its prior distribution. However, many important Bayesian bandit algorithms, such as Thompson sampling, are not arbitrary functions of their priors; rather, they select actions based on their internal posterior distribution in a relatively stable manner. We call such algorithms *n-Monte Carlo algorithms*.

Definition 3.1 (n-Monte Carlo algorithm)**.** Given $n > 0$, we say that a family of algorithms $\mathsf{alg}(\cdot)$ parameterized by $\theta \in \Theta$ is n-Monte Carlo if, for any θ, θ', step $h \geq 1$, and partial trajectory τ_{h-1},

$$\mathrm{TV}(P_{\mathsf{alg}(\theta)}(a_h \mid \tau_{h-1}) \parallel P_{\mathsf{alg}(\theta')}(a_h \mid \tau_{h-1})) \leq n \cdot \mathrm{TV}(P_\theta(\boldsymbol{\mu} \mid \tau_{h-1}) \parallel P_{\theta'}(\boldsymbol{\mu} \mid \tau_{h-1})).$$

In words, n-Monte Carlo algorithms are those Bayesian algorithms for which small changes in the posterior distribution result in small changes (up to a multiplicative factor of n) in the distribution over actions. Note that on the left-hand side, we do not need to specify the true $\theta^\star$, because each algorithm's choice of an action can only depend on τ_{h-1}. The nomenclature arises because any algorithm that selects actions based exclusively on n *samples* from its posterior $P_\theta(\boldsymbol{\mu} \mid \tau_{h-1})$ is n-Monte Carlo. However, the definition is more general and in Appendix C we describe various algorithms that satisfy the n-Monte Carlo property, summarizing key insights here:

- We show that $\mathsf{TS}(\theta)$ is 1-Monte Carlo.
- We introduce a generalization of Thompson sampling, which we call k-shot Thompson sampling (k-$\mathrm{TS}(\theta)$), that samples k means $\tilde{\boldsymbol{\mu}}_1, \ldots, \tilde{\boldsymbol{\mu}}_k$ i.i.d. from the posterior $P_\theta[\cdot \mid \tau_{h-1}]$, and selects the action $a_h \in \arg\max_a \max\{\tilde{\mu}_{1,a}, \ldots, \tilde{\mu}_{k,a}\}$. We show that k-$\mathrm{TS}(\theta)$ is k-Monte Carlo.
- We introduce a Monte Carlo approximation of the knowledge gradient algorithm [RPF12], which we call two-step Receding Horizon Control (2-$\mathrm{RHC}(\theta)$). This algorithm is non-myopic in that it chooses an action that maximizes the expected value at the subsequent time (according to its own posterior updates). We show that when $\mathcal{A}$ is finite, 2-$\mathrm{RHC}(\theta)$ is n-Monte Carlo for some n that is polynomial in $|\mathcal{A}|$ and the number of Monte Carlo samples it draws from its posterior.

3.2 General Sensitivity Upper and Lower Bounds

We are now ready to state a general prior sensitivity bound for n-Monte Carlo algorithms. For simplicity, we state our bounds for B-bounded priors, that is, $P_\theta[\mathrm{diam}(\boldsymbol{\mu}) \leq B] = 1$, and under a natural sub-Gaussian tail condition stated formally in Appendix B.2 (Theorem B.2).

Theorem 3.2. *Let $\mathsf{alg}(\cdot)$ be an n-Monte Carlo family of algorithms on horizon $H \in \mathbb{N}$, and let $\theta, \theta' \in \Theta$. Setting $\varepsilon = \mathrm{TV}(P_\theta \parallel P_{\theta'})$, we have the following guarantees.*

(a) If P_θ is B-bounded, then $|R(\theta, \mathsf{alg}(\theta)) - R(\theta, \mathsf{alg}(\theta'))| \leq 2nH^2\varepsilon \cdot B$.

(b) If P_θ is coordinate-wise σ^2-sub-Gaussian, then

$$|R(\theta, \mathsf{alg}(\theta)) - R(\theta, \mathsf{alg}(\theta'))| \leq 2nH^2\varepsilon \left(\mathrm{diam}(E_\theta[\boldsymbol{\mu}]) + \sigma \left(8 + 5\sqrt{\log\left(\frac{|\mathcal{A}|^2}{\min\{1, 2nH\epsilon\}}\right)} \right) \right).$$

Next, we complement our upper bound with a lower bound that matches Theorem 3.2(a) for n-shot Thompson sampling (an n-Monte Carlo algorithm) up to a multiplicative constant:

Theorem 3.3 (Lower Bound, Informal)**.** *For any parameter $n \in \mathbb{N}$, horizon $H \gg 1$, number of arms $N = |\mathcal{A}| \gg H$, and separation $\epsilon \ll 1/nH$, there exist two priors P_θ and $P_{\theta'}$ over bounded means $\boldsymbol{\mu} \in [0,1]^N$ such that $\mathrm{TV}(P_\theta \parallel P_{\theta'}) = \epsilon$ and*

$$R(\theta, n\text{-}\mathrm{TS}(\theta)) \geq R(\theta, n\text{-}\mathrm{TS}(\theta')) + (1 - o(1)) \cdot \frac{nH^2\epsilon}{2},$$

where the $o(1)$ decays to zero as $1/H$, H/N, $\epsilon nH \to 0$.

See Theorem D.1 for a precise, quantitative statement and Appendix D for a full proof.

Proof ideas. One of the key ingredients in the proof of Theorem 3.2, and a result which may be of independent interest, is the following bound on the total variation of the trajectory of an algorithm run with the true prior and the same algorithm run with an incorrect prior.

Proposition 3.4. *Let* $\mathsf{alg}(\cdot)$ *be an* n*-Monte Carlo family of algorithms on horizon* $H \in \mathbb{N}$*. Then,*

$$\mathrm{TV}(P_H \parallel P'_H) \leq 2nH \cdot \mathrm{TV}(P_\theta \parallel P_{\theta'}),$$

where $P_H = P_{\theta,\mathsf{alg}(\theta)}(\boldsymbol{\mu}, \tau_H)$ *and* $P'_H = P_{\theta,\mathsf{alg}(\theta')}(\boldsymbol{\mu}, \tau_H)$*.*

A full proof is given in Appendix B.5. The factor of H arises from a telescoping argument (Lemma B.7) based on the performance-difference lemma [Kak03].

For B-bounded priors, Proposition 3.4 directly translates into the sensitivity bound in Theorem 3.2(a), where the difference in rewards can be bounded as BH times the probability that the trajectory of $\mathsf{alg}(\theta)$ differs from the trajectory of $\mathsf{alg}(\theta')$. Addressing more general tail conditions like sub-Gaussianity requires more care; see Appendix B for details.

4 Meta-learning

In this section, we apply the aforementioned prior sensitivity guarantee to episodic Bayesian meta-learning and obtain sample-efficiency guarantees for canonical Bayesian bandit setups.

Suppose an episodic Bayesian meta-learner uses an exploration strategy $\mathsf{explore}^{(t)}$ in T episodes and computes an estimate $\hat{\theta} = \hat{\theta}(\tau^{(1)}, \ldots, \tau^{(T)})$ of the ground-truth parameter $\theta^\star$. Suppose further that, for any $\varepsilon, \delta \in (0, 1)$, with probability at least $1 - \delta$ over the realizations of the episodes and internal randomization of the meta-learner, the estimate $\hat{\theta}$ satisfies $\mathrm{TV}(P_{\theta^\star} \parallel P_{\hat{\theta}}) \leq \varepsilon$. Then, Theorem 3.2 implies that, for any n-Monte Carlo algorithm $\mathsf{alg}(\cdot)$, the relative performance of $\mathsf{alg}(\hat{\theta})$ compared to $\mathsf{alg}(\theta^\star)$ is (essentially) bounded as $\tilde{O}(nH^2\varepsilon)$ over horizon H.

Our task of designing meta-learners is thus reduced to that of designing estimators (and exploration strategies) for $\theta^\star$. We explain how to do so in two setups: (1) $P_{\boldsymbol{\theta}^\star}$ is a product of Beta distributions, and the rewards are Bernoulli; (2) $P_{\boldsymbol{\theta}^\star}$ is a multivariate Gaussian and the rewards are Gaussian.

4.1 Beta Priors and Bernoulli Rewards

We first consider the situation where the prior distribution is a product of Beta distributions $P_{\boldsymbol{\theta}^\star} = \bigotimes_{a \in \mathcal{A}} \mathrm{Beta}(\alpha_a^\star, \beta_a^\star)$ and the reward distribution is a product of Bernoulli distributions $\mathcal{D}(\boldsymbol{\mu}) = \bigotimes_{a \in \mathcal{A}} \mathrm{Bern}(\mu_a)$. Recall that $\mathrm{Beta}(\alpha, \beta)$ for $\alpha > 0$ and $\beta > 0$ is a continuous probability distribution supported on $(0, 1)$, and hence our parameter space Θ is the (strictly) positive orthant in $\mathbb{R}^{2|\mathcal{A}|}$.

Our approach is to directly estimate the parameters $\boldsymbol{\theta}^\star = (\boldsymbol{\alpha}^\star, \boldsymbol{\beta}^\star)$ from the observed rewards in the T episodes. Since the family of Beta distributions is an exponential family [Bro86] (with (α, β) being the natural parameters), we can appeal to general statistical theory to bound the total variation distance between two such distributions in terms of their parameter distance.

Suppose we adopt the exploration strategy where arm 1 is selected in the first n rounds in each of the first $T/|\mathcal{A}|$ episodes, arm 2 in the next $T/|\mathcal{A}|$ episodes, and so on. (We assume the horizon H and n satisfy $H \geq n \geq 2$.) We focus on the estimation of $(\alpha_1^\star, \beta_1^\star)$, as the exact same approach works for all of the arms. Let X_t denote the total reward collected in episode t. Then, the random variables $X_1, \ldots, X_{T/|\mathcal{A}|}$ are i.i.d. draws from a Beta-Binomial distribution with parameters $(\alpha_1^\star, \beta_1^\star, n)$, where n denotes the number of trials of the binomial component. The first and second moments of X_t are

$$m_1^\star = \mathbb{E}[X_t] = \frac{n\alpha_1^\star}{\alpha_1^\star + \beta_1^\star} \quad \text{and} \quad m_2^\star = \mathbb{E}[X_t^2] = \frac{n\alpha_1^\star(n(1+\alpha_1^\star) + \beta_1^\star)}{(\alpha_1^\star + \beta_1^\star)(1 + \alpha_1^\star + \beta_1^\star)}.$$

These moments uniquely determine $\alpha_1^\star$ and $\beta_1^\star$ as long as $n \geq 2$. Therefore, we can estimate $(\alpha_1^\star, \beta_1^\star)$ using plug-in estimates of the first two moments $(m_1^\star, m_2^\star)$ via the method of moments [TGG94].

Theorem 4.1. *There is an exploration strategy* $\mathsf{explore}^{(t)}$ *and an estimator* $\hat{\boldsymbol{\theta}}(\cdot)$ *with the following property. If* $P_{\boldsymbol{\theta}^\star} = \bigotimes_{a \in \mathcal{A}} \mathrm{Beta}(\alpha_a^\star, \beta_a^\star)$ *and* $\mathcal{D}(\boldsymbol{\mu}) = \bigotimes_{a \in \mathcal{A}} \mathrm{Bern}(\mu_a)$*, then there is a constant* C *depending only on* $(\boldsymbol{\alpha}^\star, \boldsymbol{\beta}^\star)$ *such that, for any* $\varepsilon, \delta \in (0, 1)$*, if* $H \geq 2$ *and*

$$T \geq \frac{C \cdot |\mathcal{A}|^2 \log(|\mathcal{A}|/\delta)}{\varepsilon^2},$$

then $\mathbb{P}[\mathrm{TV}(P_{\boldsymbol{\theta}^\star} \parallel P_{\hat{\boldsymbol{\theta}}}) \le \varepsilon] \ge 1 - \delta$.

The proof of the theorem is given in Appendix F.1.

4.2 Gaussian Priors and Gaussian Rewards

We now consider the situation where the prior distribution is a multivariate Gaussian $P_{\boldsymbol{\theta}^\star} = \mathcal{N}(\boldsymbol{\nu}_\star, \boldsymbol{\Psi}_\star)$ in $\mathbb{R}^{\mathcal{A}}$, and the reward distribution is a spherical Gaussian distribution $\mathcal{D}(\boldsymbol{\mu}) = \mathcal{N}(\boldsymbol{\mu}, \sigma^2\mathbf{I})$. Note that such a prior distribution is able to capture correlations between the arms' mean rewards in an episode, which cannot be captured by the product-form priors in the previous subsection (nor in previous work [e.g., KKZ$^+$21]).

We again directly estimate the parameters $\boldsymbol{\theta}^\star = (\boldsymbol{\nu}_\star, \boldsymbol{\Psi}_\star)$ using a simple exploration strategy and the method of moments. In each episode (which we assume have horizon $H \ge 2$), we select independent and uniformly random actions in the first two rounds. Let a_t and b_t denote the actions taken in episode t, and let r_t and s_t denote the corresponding observed rewards. Our estimates for $\boldsymbol{\nu}_\star$ and $\boldsymbol{\Psi}_\star$ based on the information collected in T episodes are[3]

$$\widehat{\boldsymbol{\nu}} := \frac{|\mathcal{A}|}{T}\sum_{t=1}^{T} r_t \mathbf{e}_{a_t} \quad \text{and} \quad \widehat{\boldsymbol{\Psi}} := \frac{|\mathcal{A}|^2}{T}\sum_{t=1}^{T} r_t s_t \left(\mathbf{e}_{a_t}\mathbf{e}_{b_t}^\top + \mathbf{e}_{b_t}\mathbf{e}_{a_t}^\top\right) - \widehat{\boldsymbol{\nu}}\widehat{\boldsymbol{\nu}}^\top.$$

Theorem 4.2. *There is an exploration strategy* $\mathsf{explore}^{(t)}$ *and an estimator* $\hat{\boldsymbol{\theta}}(\cdot)$ *with the following property. If* $P_{\boldsymbol{\theta}^\star} = \mathcal{N}(\boldsymbol{\nu}_\star, \boldsymbol{\Psi}_\star)$ *and* $\mathcal{D}(\boldsymbol{\mu}) = \mathcal{N}(\boldsymbol{\mu}, \sigma^2\mathbf{I})$*, then there is a constant* C *depending only on* $(\boldsymbol{\nu}_\star, \boldsymbol{\Psi}_\star)$ *and* σ^2 *such that, for any* $\varepsilon, \delta \in (0,1)$*, if* $H \ge 2$ *and*

$$T \ge \frac{C \cdot (|\mathcal{A}|^4 + |\mathcal{A}|^3 \log(1/\delta))}{\varepsilon^2},$$

then $\mathbb{P}[\mathrm{TV}(P_{\boldsymbol{\theta}^\star} \parallel P_{\hat{\boldsymbol{\theta}}}) \le \varepsilon] \ge 1 - \delta$.

The proof of the theorem and the precise dependence on $\boldsymbol{\nu}_\star$, $\boldsymbol{\Psi}_\star$, and σ^2 are given in Appendix F.2. The quartic dependence on $|\mathcal{A}|$ is due to estimating $\boldsymbol{\Psi}_\star$; it improves to $|\mathcal{A}|^2$ if $\boldsymbol{\Psi}_\star$ is known.

Remark 1 (Comparison to [KKZ$^+$21])**.** [KKZ$^+$21] study the case where $P_{\boldsymbol{\theta}^\star} = \mathcal{N}(\boldsymbol{\nu}_\star, \sigma_0^2\mathbf{I})$, which is a product-form prior over means $\boldsymbol{\mu}$ with known σ_0^2. For $\tilde{\epsilon} = |\mathcal{A}| \cdot \|\boldsymbol{\nu}_\star - \widehat{\boldsymbol{\nu}}\|_\infty / \sigma_0$, they show that[4]

$$|R(\boldsymbol{\theta}, \mathsf{TS}(\boldsymbol{\theta})) - R(\boldsymbol{\theta}, \mathsf{TS}(\hat{\boldsymbol{\theta}}))| \le \mathcal{O}(\|\boldsymbol{\nu}_\star\|_\infty + \sigma_0\sqrt{\log(H/\tilde{\epsilon})}) \cdot H^2\tilde{\epsilon}.$$

On the other hand, Theorem 3.2 applied to the 1-Monte Carlo Thompson Sampling algorithm (and bounding $\mathrm{diam}(\boldsymbol{\nu}_\star) \le \|\boldsymbol{\nu}_\star\|_\infty$) yields the same inequality, but with $\tilde{\epsilon}$ replaced by $\epsilon = \mathrm{TV}(\mathcal{N}(\boldsymbol{\nu}_\star, \sigma_0^2 I) \parallel \mathcal{N}(\widehat{\boldsymbol{\nu}}, \sigma_0^2 I)) \le \|\boldsymbol{\nu}_\star - \widehat{\boldsymbol{\nu}}\|_2/\sigma_0$. Note that $\tilde{\epsilon}$ is always larger than ϵ by a factor of at least $\sqrt{|\mathcal{A}|}$; thus, our result is strictly sharper.

5 Experiments

We demonstrate the generality of our results in three distinct meta-learning experimental settings. First, we study a simple multi-arm bandit scenario with Gaussian prior and Gaussian rewards, where we demonstrate how meta-learning higher-order moments of the prior can significantly improve performance. Next, we consider a Gaussian linear contextual bandits scenario, to demonstrate the generality of Bayesian meta-learning. Finally, we study a more interesting multi-arm bandit problem with discrete priors, where, in addition to the value of meta-learning, we see that look-ahead algorithms can substantially outperform Thompson sampling. Additional experimental details are presented in Appendix A.

Gaussian MAB. Our first scenario is a multi-arm bandit problem with Gaussian prior and Gaussian reward. The instance has $|\mathcal{A}| = 6$ arms and each episode has horizon $H = 10$. The prior is $\mathcal{N}(\boldsymbol{\nu}_\star, \boldsymbol{\Psi}_\star)$ where $\boldsymbol{\nu}_\star = [0.5, 0, 0, 0.1, 0, 0]$ and $\boldsymbol{\Psi}_\star$ has block structure so that arms $1, 2, 3$ are highly

[3] This estimator can be generalized to explore for more of the episode and use more of the observed rewards.

[4] The following optimizes Lemma 5 of [KKZ$^+$21] over its free parameter $\delta > 0$ for $\tilde{\epsilon}$ small.

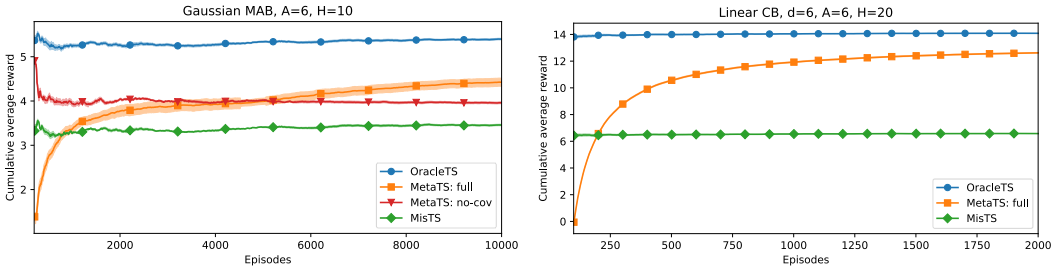

Figure 1: Learning curves for Gaussian MAB and linear CB experiments. We run 100 replicates per algorithm and visualize two standard errors with error bands. For meta-learners we tune the number of exploration rounds and display the performance of the best configuration at each point, which we call the upper envelope.

correlated, and analogously for arms $4, 5, 6$. The rewards are Gaussian with covariance 1, which is known to all learners.

We run four algorithms. Two are non-meta-learning Thompson sampling algorithms: **`OracleTS`**, which uses the correct prior, and **`MisTS`**, which uses the misspecified prior $\mathcal{N}(\mathbf{0}, \mathbf{I})$. We also run **`MetaTS:no-cov`** which only attempts to meta-learn the prior mean μ_0 and assumes that the prior covariance matrix is the identity (this algorithm is essentially the one studied in [KKZ+21]). Finally, our algorithm is **`MetaTS:full`** which meta-learns both the prior mean and covariance. Both meta-learners are run in an explore-then-commit fashion where the first T_0 episodes are used for exploration.[5]

In Figure 1, we plot the cumulative average per-episode reward for each algorithm, where for the meta-learners we sweep over many choices of T_0 and display the pointwise best (i.e., the upper envelope). The experiment clearly shows the value of meta-learning as both **`MetaTS:no-cov`** and **`MetaTS:full`** quickly outperform misspecified TS. Additionally, we also see the importance of learning the covariance matrix, even though it can require many samples. Indeed, the final performance of **`MetaTS:full`** with $T_0 = 5$K, ignoring the regret incurred due to exploration, is competitive with **`OracleTS`**, while **`MetaTS:no-cov`** asymptotes to a much lower performance (see Figure 3 in Appendix A).

Gaussian linear contextual bandits. Our second experiment concerns Gaussian linear contextual bandits. Here we run **`OracleTS`**, **`MisTS`**, and **`MetaTS:full`**, on a synthetic linear contextual bandit problem where there are $|\mathcal{A}| = 6$ actions each with a $d = 6$ dimensional action feature (generated stochastically at each time step), and with horizon $H = 20$. The prior is over the linear parameter $\boldsymbol{\mu}$ that determines the reward for action-feature $x_a \in \mathbb{R}^d$ as $r(a) \sim \mathcal{N}(\langle \boldsymbol{\mu}, \mathbf{x}_a \rangle, 1)$. We set the prior as $\mathcal{N}(\mathbf{1}, \boldsymbol{\Psi}_\star)$ where $\boldsymbol{\Psi}_\star$ is a scaled-down version of the block diagonal matrix used in the previous experiment. In the right panel of Figure 1 we again see that by meta-learning the prior, we quickly outperform the misspecified approach and asymptotically achieve the oracle performance. This demonstrates that Bayesian meta-learning is quite broadly applicable and highlights the importance of our general theoretical development.

Discrete bandits. Finally, we study a synthetic MAB setting with $|\mathcal{A}| = 20$ arms and a prior supported on a finite set of 16 reward distributions (tasks), under each of which rewards are deterministic. The instance is visualized in the left panel of Figure 2. It is constructed so that each task has a unique optimal arm and there are four arms that can quickly identify which task the agent is in (arms A1, A6, A11, A16), so that it can infer the optimal arm. Additionally, the prior is concentrated on the first four tasks, so that pulling the first identifying arm almost always reveals the current task.

We evaluate 6 algorithms: Oracle, Misspecified, and Meta-learning each with TS and Monte-Carlo Knowledge Gradient (an instantiation of the n-Monte Carlo 2-RHC(θ) algorithm mentioned in Section 3.1 and detailed in Appendix C) as the base learners, and we visualize the results in the top right panel of Figure 2. Perhaps more revealing is the bottom right panel of Figure 2, where we visualize the empirical distribution over the first arm pull in each episode for each algorithm. We see that **`OracleTS`** typically plays uniformly over arms A2–A5 in the first round as these are highly likely

[5]For **`MetaTS:no-cov`**, we follow [KKZ+21] and only use the first step of each exploration episode for exploration, switching to TS with the current prior estimate for the rest of the episode. On the other hand, **`MetaTS:full`** explores for all time steps in the first T_0 episodes.

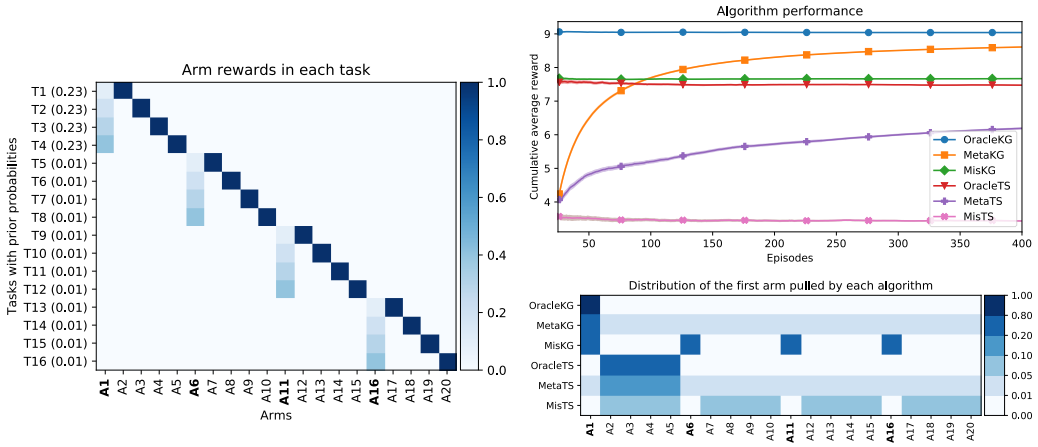

Figure 2: Synthetic experiments with discrete MAB for $|\mathcal{A}| = 20$ and $H = 10$. *Left*: visualization of the instance showing the reward for each of the arms in each of the 16 possible tasks along with the prior distribution over tasks (probabilities rounded, actual values are $9/40$ and $1/120$). *Top right*: learning curves for 6 algorithms (100 replicates, error bands at 2 standard errors, we tune the number of exploration rounds and plot upper envelopes for meta-learners). *Bottom right*: empirical distribution of the first arm pulled in each episode by each algorithm. Note that the color scale is non-linear.

to be the optimal arm under the prior, while **`MisTS`** plays uniformly over the 16 plausibly optimal arms. **`MetaTS`** quickly learns to play uniformly over arms A2–A5 and is asymptotically competitive with **`OracleTS`**.

Playing the identifying arms is crucial for optimal behavior. However, since TS is myopic and these arms never produce large rewards, TS will never play them. Thus, to achieve optimal behavior, we must use a less myopic base learner like Knowledge Gradient. As can be seen, both **`OracleKG`** and **`MisKG`** first play the identifying arms, where the oracle almost always pulls the first one while **`MisKG`** plays them uniformly. The performance of **`OracleKG`** is much better than all TS configurations. Finally, the meta-learning configuration of Knowledge Gradient quickly learns to pull the first identifying arm and competes with **`OracleKG`**.

6 Discussion

In our simulations, we demonstrated the superiority of more expressive prior families (e.g., modeling mean-covariances) and non-myopic base algorithms (e.g., Knowledge Gradient) over less expressive priors (e.g., product measures) and greedy base learners (e.g., Thompson sampling). Notably, the generality and flexibility of our theoretical contributions ensure robustness to prior misspecification even for these richer priors and sophisticated base learners.

Still, theory and experiments alike point to a tradeoff: despite the potential for improved performance, richer prior families are harder to learn, and some base learners (e.g., n-Monte Carlo algorithms for large n) can be more sensitive to incorrect priors. It is an exciting direction for future work to investigate the joint problems of *model selection* (over priors) and *algorithm selection* (over base learners) in order to optimally navigate these tradeoffs. Perhaps model and algorithm selection can be coupled so that certain base learners exhibit improved performance, or greater robustness, over certain classes of priors. We would like to further understand how these tradeoffs interface with computational burdens of using certain priors and base learners, and whether our sensitivity analysis extends to computationally efficient approximations of sampling-based decision-making algorithms (e.g., via Laplace approximations, MCMC, Gibbs Sampling, and Variational Methods; the long-horizon performance of Thompson sampling under approximate inference has already been studied [PAYD19]). Finally, we hold hope that a more instance-dependent analysis may improve our sensitivity bounds for certain families of priors, which may in turn inform more clever exploration strategies that circumvent worst-case tradeoffs.

References

[AG12] Shipra Agrawal and Navin Goyal. Analysis of thompson sampling for the multi-armed bandit problem. In *Conference on Learning Theory*, pages 1–39, 2012.

[AL17] Marc Abeille and Alessandro Lazaric. Linear thompson sampling revisited. In *Artificial Intelligence and Statistics*, pages 176–184. PMLR, 2017.

[AL18] Marc Abeille and Alessandro Lazaric. Improved regret bounds for thompson sampling in linear quadratic control problems. In *International Conference on Machine Learning*, pages 1–9. PMLR, 2018.

[ALB13] Mohammad Gheshlaghi Azar, Alessandro Lazaric, and Emma Brunskill. Sequential transfer in multi-armed bandit with finite set of models. *arXiv preprint arXiv:1307.6887*, 2013.

[AYS15] Yasin Abbasi-Yadkori and Csaba Szepesvári. Bayesian optimal control of smoothly parameterized systems. In *UAI*, pages 1–11. Citeseer, 2015.

[Bax98] Jonathan Baxter. Theoretical models of learning to learn. In *Learning to learn*, pages 71–94. Springer, 1998.

[Bax00] Jonathan Baxter. A model of inductive bias learning. *Journal of artificial intelligence research*, 12:149–198, 2000.

[BMD+05] Arindam Banerjee, Srujana Merugu, Inderjit S Dhillon, Joydeep Ghosh, and John Lafferty. Clustering with bregman divergences. *Journal of Machine Learning Research*, 6(10), 2005.

[BMP+94] James O Berger, Elías Moreno, Luis Raul Pericchi, M Jesús Bayarri, José M Bernardo, Juan A Cano, Julián De la Horra, Jacinto Martín, David Ríos-Insúa, Bruno Betrò, et al. An overview of robust bayesian analysis. *Test*, 3(1):5–124, 1994.

[Bro86] Lawrence D Brown. *Fundamentals of Statistical Exponential Families with Applications in Statistical Decision Theory*, volume 9 of *Lecture Notes-Monograph Series*. Institute of Mathematical Statistics, 1986.

[CLP20] Leonardo Cella, Alessandro Lazaric, and Massimiliano Pontil. Meta-learning with stochastic linear bandits. In *International Conference on Machine Learning*, pages 1360–1370. PMLR, 2020.

[DSC+16] Yan Duan, John Schulman, Xi Chen, Peter L Bartlett, Ilya Sutskever, and Pieter Abbeel. Rl^2: Fast reinforcement learning via slow reinforcement learning. *arXiv preprint arXiv:1611.02779*, 2016.

[GMM14] Aditya Gopalan, Shie Mannor, and Yishay Mansour. Thompson sampling for complex online problems. In *International Conference on Machine Learning*, pages 100–108. PMLR, 2014.

[GMPT16] Mohammad Ghavamzadeh, Shie Mannor, Joelle Pineau, and Aviv Tamar. Bayesian reinforcement learning: A survey. *arXiv preprint arXiv:1609.04436*, 2016.

[HGH+19] Jan Humplik, Alexandre Galashov, Leonard Hasenclever, Pedro A Ortega, Yee Whye Teh, and Nicolas Heess. Meta reinforcement learning as task inference. *arXiv preprint arXiv:1905.06424*, 2019.

[HYC01] Sepp Hochreiter, A Steven Younger, and Peter R Conwell. Learning to learn using gradient descent. In *International Conference on Artificial Neural Networks*, pages 87–94. Springer, 2001.

[Kak03] Sham Machandranath Kakade. *On the sample complexity of reinforcement learning*. PhD thesis, UCL (University College London), 2003.

[KKM12] Emilie Kaufmann, Nathaniel Korda, and Rémi Munos. Thompson sampling: An asymptotically optimal finite-time analysis. In *International conference on algorithmic learning theory*, pages 199–213, 2012.

[KKZ+21] Branislav Kveton, Mikhail Konobeev, Manzil Zaheer, Chih-wei Hsu, Martin Mladenov, Craig Boutilier, and Csaba Szepesvari. Meta-thompson sampling. *arXiv preprint arXiv:2102.06129*, 2021.

[Lin02] Torgny Lindvall. *Lectures on the coupling method*. Courier Corporation, 2002.

[LL16] Che-Yu Liu and Lihong Li. On the prior sensitivity of thompson sampling. In *International Conference on Algorithmic Learning Theory*, pages 321–336, 2016.

[OVR17] Ian Osband and Benjamin Van Roy. Why is posterior sampling better than optimism for reinforcement learning? In *International Conference on Machine Learning*, pages 2701–2710. PMLR, 2017.

[PAYD19] My Phan, Yasin Abbasi Yadkori, and Justin Domke. Thompson sampling and approximate inference. *Advances in Neural Information Processing Systems*, 32:8804–8813, 2019.

[RPF12] Ilya O Ryzhov, Warren B Powell, and Peter I Frazier. The knowledge gradient algorithm for a general class of online learning problems. *Operations Research*, 60(1):180–195, 2012.

[Rus16] Daniel Russo. Simple bayesian algorithms for best arm identification. In *Conference on Learning Theory*, pages 1417–1418. PMLR, 2016.

[RVR16] Daniel Russo and Benjamin Van Roy. An information-theoretic analysis of thompson sampling. *The Journal of Machine Learning Research*, 17(1):2442–2471, 2016.

[Str00] Malcolm Strens. A bayesian framework for reinforcement learning. In *ICML*, volume 2000, pages 943–950, 2000.

[SWDN09] Songsak Sriboonchita, Wing-Keung Wong, Sompong Dhompongsa, and Hung T Nguyen. *Stochastic dominance and applications to finance, risk and economics*. CRC Press, 2009.

[TGG94] Ram C Tripathi, Ramesh C Gupta, and John Gurland. Estimation of parameters in the beta binomial model. *Annals of the Institute of Statistical Mathematics*, 46(2):317–331, 1994.

[Tho33] William R Thompson. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika*, 25(3/4):285–294, 1933.

[Thr96] Sebastian Thrun. Explanation-based neural network learning. In *Explanation-Based Neural Network Learning*, pages 19–48. Springer, 1996.

[Thr98] Sebastian Thrun. Lifelong learning algorithms. In *Learning to learn*, pages 181–209. Springer, 1998.

[Ver18] Roman Vershynin. *High-dimensional probability: An introduction with applications in data science*, volume 47. Cambridge University Press, 2018.

[WKNT+16] Jane X Wang, Zeb Kurth-Nelson, Dhruva Tirumala, Hubert Soyer, Joel Z Leibo, Remi Munos, Charles Blundell, Dharshan Kumaran, and Matt Botvinick. Learning to reinforcement learn. *arXiv preprint arXiv:1611.05763*, 2016.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] We describe in discussion the limitations to n-Monte Carlo algorithms, computational challenges of sampling, and statistical challenges in estimating complex priors.
   (c) Did you discuss any potential negative societal impacts of your work? [N/A]
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes] See, e.g. Definition 3.1
   (b) Did you include complete proofs of all theoretical results? [Yes] Yes, in the supplement.
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] See supplement for code, and Appendix A for experimental details.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] See supplement for code, and Appendix A for experimental details.
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] see, e.g. Figure 1.
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] Yes, see Appendix A in supplement.
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [N/A]
   (b) Did you mention the license of the assets? [N/A]
   (c) Did you include any new assets either in the supplemental material or as a URL? [N/A]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]