Non-Compliant Bandits

Branislav Kveton
bkveton@amazon.com
Amazon
United States

Yi Liu
yiam@amazon.com
Amazon
United States

Johan Matteo Kruijssen
matteok@amazon.com
Amazon
United States

Yisu Nie
yisnie@amazon.com
Amazon
United States

ABSTRACT

Bandit algorithms arose as a standard approach to learning better models online. As they become more popular, they are increasingly deployed in complex machine learning pipelines, where their actions can be overwritten. For example, in ranking problems, a list of recommended items can be modified by a downstream algorithm to increase diversity. This may break the classic bandit algorithms and lead to linear regret. Specifically, if the proposed action is not taken, uncertainty in its estimated mean reward may not get reduced. In this work, we study this setting and call it non-compliant bandits; as the agent tries to learn rewarding actions that comply with a downstream task. We propose two algorithms, compliant contextual UCB (`CompUCB`) and Thompson sampling (`CompTS`), which learn separate reward and compliance models. The compliance model allows the agent to avoid non-compliant actions. We derive a sublinear regret bound for `CompUCB`. We also conduct experiments that compare our algorithms to classic bandit baselines. The experiments show failures of the baselines and that we mitigate them by learning compliance models.

CCS CONCEPTS

• **Theory of computation** → **Online learning algorithms**; **Reinforcement learning**; **Active learning**.

KEYWORDS

bandits; exploration; online learning; reinforcement learning; active learning

ACM Reference Format:
Branislav Kveton, Yi Liu, Johan Matteo Kruijssen, and Yisu Nie. 2023. Non-Compliant Bandits. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management (CIKM '23), October 21–25, 2023, Birmingham, United Kingdom.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3583780.3614990

CIKM '23, October 21–25, 2023, Birmingham, United Kingdom

ACM ISBN 979-8-4007-0124-5/23/10.
https://doi.org/10.1145/3583780.3614990

1 INTRODUCTION

A bandit [5, 23, 27] is a framework for sequential decision making under uncertainty where the learning agent takes actions, potentially based on context, and observes rewards. The typical goal of the agent is to learn an optimal policy that maximizes the expected cumulative reward, which equivalently minimizes the expected cumulative regret. Many types of bandit algorithms exist: contextual [1, 4, 10], combinatorial [7, 15, 20, 21, 34], and even for multiple objectives [31]. The underlying principle in all of them is that the uncertainty in the mean reward estimate of the proposed action is reduced after the action is taken. When the taken action differs from the proposed action, the algorithms can have linear regret. In this work, we study this setting under the name of non-compliant bandits.

An action is *non-compliant* if it is overwritten or censored later. Non-compliance is common in practice. For instance, an upstream model may recommend a movie that a downstream model or rule-based system overwrites. More specifically, the downstream system may use a propensity model to predict the probability of violence and overwrite violent recommendations for young viewers. When the upstream model does not comply, it may or may not know which movie was actually recommended [6, 36]. Recommendations of an upstream model may also be censored. For instance, a voice assistant may suppress recommendations that could disturb the user. Since bandit algorithms are a popular approach to learning to recommend [19, 20, 28, 34, 41], where action overwriting and censorship are common, we set out to study non-compliance in the bandit setting.

Only three prior works studied a similar setting [17, 33, 38]. In the first two, the agent observes the taken action, similarly to our work. Stirn and Jebara [38] do not make this assumption. However, they model reward and compliance separately, similarly to our work. While the first two works are non-contextual, the last one considers a trivial tabular case, with a separate Bernoulli bandit per context. In summary, non-compliant bandit algorithms that could handle realistic context, and thus be practical, do not exist. Since it is important to integrate bandit algorithms with modern machine learning pipelines, the problem of non-compliant bandits is significantly understudied, and we attempt to fill this gap.

In this work, we propose two non-compliant bandit algorithms: *compliant contextual Thompson sampling* (`CompTS`) and *compliant contextual upper confidence bound* (`CompUCB`). The algorithms act optimistically with respect to the product of the compliance probability and mean reward. Both the reward and compliance models are contextual and learned separately. We focus on linear reward and logistic compliance models. This leads to simple practical algorithms that are general, due to using features; and both computationally and statistically efficient, due to relying on linear and generalized linear models. We prove a $\tilde{O}(d\sqrt{n})$ regret bound for CompUCB with d features over n rounds. We experiment with both synthetic and real-world problems, where CompUCB and CompTS are compared to classic bandit baselines and a neural network bandit algorithm. Our experiments show that CompUCB and CompTS have much lower regret than their classic counterparts, and are more efficient than learning a neural network reward model.

This paper is organized as follows. In Section 2, we introduce the setting of non-compliant bandits. In Section 3, we propose CompTS and CompUCB for non-compliant bandits. In Section 4, we prove a $\tilde{O}(d\sqrt{n})$ regret bound for CompUCB with d features over n rounds. In Section 5, we evaluate CompTS and CompUCB on both synthetic and real-world problems. We conclude in Section 6.

2 SETTING

We adopt the following notation. Random variables are capitalized, except for Greek letters like θ. For any positive integer n, we define $[n] = \{1, \dots, n\}$. The indicator function is $\mathbb{1}\{\cdot\}$. The i-th entry of vector v is v_i. If the vector is already indexed, such as v_j, we write $v_{j,i}$. We denote the maximum and minimum eigenvalues of matrix $M \in \mathbb{R}^{d \times d}$ by $\lambda_1(M)$ and $\lambda_d(M)$, respectively.

A *contextual bandit* [1, 4, 25, 28] is a sequential decision-making problem where the mean rewards of actions depend on context. We model it as follows. A learning agent interacts with the environment for n rounds. In round $t \in [n]$, it takes an action $A_t \in \mathcal{A}_t$ and then observes its stochastic reward $Y_t \in \mathbb{R}$, where $\mathcal{A}_t \subseteq \mathcal{A}$ is a *round-dependent action set* and $\mathcal{A}$ is the set of all actions. We assume that $\mathcal{A} \subseteq \mathbb{R}^d$ is a set of d-dimensional feature vectors. Since $\mathcal{A}_t$ depends on t, we can encode any round-dependent context in it, such as all feasible movies to recommend in round t. The mean reward of action $a \in \mathcal{A}$ is $f(a; \theta_*)$, where $f : \mathcal{A} \times \Theta \to \mathbb{R}$ is the *reward function*, $\theta_* \in \Theta$ is an unknown *reward parameter*, and Θ is the set of feasible reward parameters. The stochastic reward of action A_t in round t is $Y_t = f(A_t; \theta_*) + \varepsilon_t$, where ε_t is independent σ^2-sub-Gaussian noise.

2.1 Many Flavors of Non-Compliance

Our bandit model can be viewed as a contextual bandit where the taken action may differ from the proposed action A_t. We denote the taken action by $\tilde{A}_t$, and relate the actions as $\tilde{A}_t = \kappa(A_t)$, for some function κ. The function κ may be round-dependent and stochastic, and is unknown to the agent. The reward of action $\tilde{A}_t$ in round t is $\tilde{Y}_t = f(\tilde{A}_t; \theta_*) + \tilde{\varepsilon}_t$, where $\tilde{\varepsilon}_t$ is independent σ^2-sub-Gaussian noise. In the rest of this section, we discuss two variants of the problem, where $\tilde{A}_t$ is either observed or not. In both variants, $\tilde{Y}_t$ is observed, since learning without feedback would be impossible.

Single model. Suppose that the taken action $\tilde{A}_t$ is unobserved or we choose not to model it. Then our problem can be still solved as a contextual bandit where the mean reward of action a is $f(\kappa(a); \theta_*)$, and it is learned from pairs $(A_t, \tilde{Y}_t)$. We call this approach a *single model*. The challenge is that $f(\kappa(a); \theta_*)$ may be a complex function of action a, representing how the downstream model replaces a with another action. These models are often complex, and thus arguably hard to mimic and learn.

Classic model. Now suppose that the taken action $\tilde{A}_t$ is observed. Then any contextual bandit algorithm can be used to learn $f(a; \theta_*)$ from pairs $(\tilde{A}_t, \tilde{Y}_t)$. We call this approach *classic*. It is not immediately clear that this approach may fail. The reason is that when an action A_t is proposed but the model is updated with $(\tilde{A}_t, \tilde{Y}_t)$, the uncertainty in the mean reward estimate of action A_t may not be reduced. Therefore, a classic bandit algorithm may repeatedly take the same action whose uncertainty is not reduced and get stuck.

To address the shortcomings of existing bandit algorithms, we propose modeling compliance. At a high level, we learn a function $g(a) = \mathbb{P}(\kappa(a) = a)$, which represents the probability that action a complies. This is possible when both the proposed A_t and taken $\tilde{A}_t$ actions are observed. This prevents the agent from taking actions whose uncertainty cannot be reduced, which leads to failures of the classic bandit algorithms.

2.2 Non-Compliant Bandits

A *non-compliant bandit* is a variant of a contextual bandit defined as follows. In round $t \in [n]$, the agent proposes an action $A_t \in \mathcal{A}_t$ and gets two observations. The first observation is the *taken action* $\tilde{A}_t \in \mathcal{A}_t$. This action may differ from the *proposed action* A_t. The second observation is the *stochastic reward* of the taken action $\tilde{A}_t$, $\tilde{Y}_t = f(\tilde{A}_t; \theta_*) + \tilde{\varepsilon}_t$, where $\tilde{\varepsilon}_t$ is independent σ^2-sub-Gaussian noise. We discuss other feedback models in Section 2.1.

We consider a probabilistic compliance model, which models the probability that an action complies. Specifically, the probability that action $a \in \mathcal{A}$ complies is $g(a; \psi_*)$, where $g : \mathcal{A} \times \Psi \to [0, 1]$ is the *compliance function*, $\psi_* \in \Psi$ is an unknown *compliance parameter*, and Ψ is the set of feasible compliance parameters. Since the agent knows both A_t and $\tilde{A}_t$, it effectively observes a stochastic *compliance indicator* $C_t = \mathbb{1}\left\{A_t = \tilde{A}_t\right\}$, if the proposed and taken actions are the same. This feedback relates to the compliance probability as $C_t = g(A_t; \psi_*) + \eta_t$, where η_t is independent Bernoulli noise.

Now we discuss the notion of optimality. An important point to realize is that the classic notion of regret cannot be minimized. As an example, consider a bandit problem where the optimal action is always overwritten by the second best action. Therefore, in this work, we define the *optimal action* in round t as the one with the highest mean reward weighted by its compliance probability,

$$A_{t,*} = \arg\max{}_{a \in \mathcal{A}_t} \, g(a; \psi_*) f(a; \theta_*) \,.$$

In plain English, this is the highest mean reward that the agent can attain without modeling non-compliance, when $C_t = 0$. Specifically, suppose that $f(a; \theta_*) \geq 0$ for all actions $a \in \mathcal{A}$, and that the reward and compliance noise are independent. Then

$$\begin{aligned}
\mathbb{E}\left[\tilde{Y}_t \,\middle|\, A_t = a\right] &= \mathbb{E}\left[C_t \tilde{Y}_t + (1 - C_t)\tilde{Y}_t \,\middle|\, A_t = a\right] \\
&= \mathbb{E}\left[C_t Y_t + (1 - C_t)\tilde{Y}_t \,\middle|\, A_t = a\right] \\
&\geq \mathbb{E}\left[C_t Y_t \,\middle|\, A_t = a\right] = g(a; \psi_*) f(a; \theta_*) \,.
\end{aligned}$$

As discussed earlier, the downstream algorithm is usually complex; and thus hard to mimic and learn. So $g(a; \psi_*) f(a; \theta_*)$ is a reasonable measure of attainable reward. The corresponding *expected n-round*

Algorithm 1 `CompTS`: Compliant contextual TS.

```
Explore for $\tau$ rounds
for $t = \tau + 1, \ldots, n$ do
    Update distributions $P_{t,\theta}$ and $P_{t,\psi}$
    Sample $\theta_t \sim P_{t,\theta}$ and $\psi_t \sim P_{t,\psi}$
    Propose action $A_t \leftarrow \arg\max_{a \in \mathcal{A}_t} g(a; \psi_t) f(a; \theta_t)$
    Action $\tilde{A}_t$ is taken and $\tilde{Y}_t$ is observed
```

Algorithm 2 `CompUCB`: Compliant contextual UCB.

```
Explore for $\tau$ rounds
for $t = \tau + 1, \ldots, n$ do
    Update UCBs $U_{t,\theta}$ and $U_{t,\psi}$
    Propose action $A_t \leftarrow \arg\max_{a \in \mathcal{A}_t} U_{t,\psi}(a) U_{t,\theta}(a)$
    Action $\tilde{A}_t$ is taken and $\tilde{Y}_t$ is observed
```

regret is

$$R(n) = \mathbb{E}\left[\sum_{t=1}^{n} g(A_{t,*}; \psi_*) f(A_{t,*}; \theta_*) - g(A_t; \psi_*) f(A_t; \theta_*)\right] .$$

3 ALGORITHMS

The key idea in our algorithms is to act optimistically with respect to the product of the compliance probability and mean reward. One natural property of this design is that the agent ultimately learns to take compliant actions, because the optimistic overestimates of the compliance probability of non-compliant actions eventually go to zero. When this happens, the agents starts taking compliant actions. While these may have lower mean rewards, they can be higher in expectation when the compliance is considered.

3.1 Algorithm Designs

We consider two algorithm designs. The first algorithm is a form of contextual *Thompson sampling (TS)* [2–4, 22, 35, 39]. We present it in Algorithm 1 and call it `CompTS`, which stands for *compliant contextual TS.* The key idea in `CompTS` is to sample the unknown model parameters from their posterior distributions and then act optimistically with respect to them. The distribution of θ_* in round t is denoted by $P_{t,\theta}$ and θ_t is sampled from it. The distribution of ψ_* in round t is denoted by $P_{t,\psi}$ and ψ_t is sampled from it. The posteriors are detailed in Sections 3.2 and 3.3. Since our setting is frequentist (Section 2), posterior sampling only serves as a randomization oracle that is optimistic with a sufficiently high probability [4].

The second algorithm relies on *upper confidence bounds (UCBs)* [1, 10, 14, 29]. We present it in Algorithm 2 and call it `CompUCB`, which stands for *compliant contextual UCB.* The key idea is to act optimistically with respect to the product of estimated compliance probabilities and mean rewards. The UCB on the mean reward of action a in round t is $U_{t,\theta}(a)$. The UCB on the compliance probability of action a in round t is $U_{t,\psi}(a)$. Note that when the mean rewards are non-negative, $U_{t,\psi}(a)U_{t,\theta}(a)$ is a valid UCB on $g(a; \psi_*) f(a; \theta_*)$. The UCBs are detailed in Sections 3.2 and 3.3.

To have computationally-efficient implementations of `CompTS` and `CompUCB`, we consider specific reward and compliance models. In particular, the reward function is *linear*, $f(a; \theta) = a^\top \theta$ for any $a \in \mathcal{A}$; and the compliance function is *logistic*, $g(a; \psi) = \mu(a^\top \psi)$ for any $a \in \mathcal{A}$, where $\mu(x) = 1/(1 + \exp[-x])$ is a *sigmoid.* We make these choices for two reasons. First, both models are very general and flexible, because any function can be approximated by a linear function of non-linear features, which can be engineered based on domain knowledge or previously logged data. Second, exploration with linear and logistic models is well understood. Therefore, we can build on existing techniques, both in the algorithm design and analysis [1, 2, 10, 14, 22, 29].

3.2 Reward Model Estimation

Since the reward of the proposed action A_t may not be observed, when the action is non-compliant, we estimate the reward model from taken actions $\tilde{A}_t$ and their observations $\tilde{Y}_t$. Specifically, the reward parameter in round t is estimated using regularized least squares as

$$\hat{\theta}_t = \sigma^{-2} G_{t,\theta}^{-1} \sum_{\ell=1}^{t-1} \tilde{A}_\ell \tilde{Y}_\ell \,, \quad G_{t,\theta} = \lambda I_d + \sigma^{-2} \sum_{\ell=1}^{t-1} \tilde{A}_\ell \tilde{A}_\ell^\top \,, \qquad (1)$$

where $G_{t,\theta}$ is the Gram matrix for parameter θ_* in round t and $\lambda > 0$ is a regularization parameter. This design is motivated by `LinUCB` [1]. Note that both $\hat{\theta}_t$ and $G_{t,\theta}$ can be updated online in $O(d^2)$ time using the Sherman-Morrison formula.

An upper confidence bound that holds jointly over all rounds with probability $1 - \delta$ is

$$U_{t,\theta}(a) = a^\top \hat{\theta}_t + c_\theta \|a\|_{\hat{\Sigma}_{t,\theta}} \,, \qquad (2)$$

where $\hat{\Sigma}_{t,\theta} = G_{t,\theta}^{-1}$ and $c_\theta = O(\sqrt{d \log(1/\delta)})$. We state c_θ precisely in the proof of Theorem 1. The posterior distribution is a multivariate Gaussian centered at $\hat{\theta}_t$, $P_{t,\theta}(\cdot) = \mathcal{N}(\cdot; \hat{\theta}_t, \hat{\Sigma}_{t,\theta})$.

3.3 Compliance Model Estimation

The compliance model is logistic, $g(a; \psi_*) = \mu(a^\top \psi_*)$, where μ is a sigmoid. We estimate it from proposed actions A_t and their compliance indicator $C_t = \mathbb{1}\{A_t = \tilde{A}_t\}$. More specifically, the compliance parameter in round t is estimated using logistic regression as

$$\hat{\psi}_t = \arg\max_{\psi \in \Psi} \prod_{\ell=1}^{t-1} g(A_\ell; \psi)^{C_\ell} (1 - g(A_\ell; \psi))^{1 - C_\ell} \,. \qquad (3)$$

We solve this problem by *iteratively reweighted least squares (IRLS)* [40]. This is an iterative algorithm and each of its iterations takes $O(t)$ time, since (3) contains $O(t)$ terms. To speed up the computation, we initialize IRLS in round t with a solution from round $t - 1$. After that, IRLS typically converges in a single step. This speedup is simple and sufficient for our needs. Other works on generalized linear bandits could be used to increase computational efficiency. For instance, Jun et al. [16] apply the online Newton step and Ding et al. [12] apply online stochastic gradient descent.

Uncertainty in the estimated compliance parameter $\hat{\psi}_t$ is modeled using the Gram matrix $G_{t,\psi} = \sum_{\ell=1}^{t-1} A_\ell A_\ell^\top$, which can be updated online in $O(d^2)$ time. An upper confidence bound that holds jointly over all rounds with probability $1 - \delta$ is

$$U_{t,\psi}(a) = \mu(a^\top \hat{\psi}_t + c_\psi \|a\|_{\hat{\Sigma}_{t,\psi}}) \,, \qquad (4)$$

where $\hat{\Sigma}_{t,\psi} = G_{t,\psi}^{-1}$ and $c_\psi = O(\sqrt{d \log(1/\delta)})$. This design is based on Li et al. [29], who compute an upper confidence bound on $a^\top \psi_*$ instead of $\mu(a^\top \psi_*)$. We state c_ψ in the proof of Theorem 1.

The posterior distribution is a multivariate Gaussian centered at the estimated compliance parameter $\hat{\psi}_t$,

$$P_{t,\psi}(\cdot) = \mathcal{N}(\cdot; \hat{\psi}_t, \hat{\Sigma}_{t,\psi}) .$$

Finally, note that (3) and $\hat{\Sigma}_{t,\psi}$ are ill defined unless $\lambda_d(G_{t,\psi}) > 0$. To guarantee that this condition holds, we initially explore randomly in CompUCB and CompTS until $\lambda_d(G_{t,\psi}) \geq 1$. Our analysis is also under the assumption that $\lambda_d(G_{t,\psi}) \geq 1$.

4 ANALYSIS

In this section, we analyze CompUCB and make the following assumptions. First, the feature vectors $a \in \mathcal{A}$ are bounded as $\|a\|_2 \leq 1$. This assumption is standard and without loss of generality. Second, we assume that $f(a; \theta_*) \in [0, 1]$ for all $a \in \mathcal{A}$. This can be satisfied by rescaling the original function and thus is without loss of generality. Under this assumption, $g(a; \psi_*) f(a; \theta_*) \in [0, 1]$, which we use in the proof. We also assume that $U_{t,\theta}(a) \leq 1$, which holds when $U_{t,\theta}(a)$ is clipped at 1. We have $U_{t,\psi}(a) \leq 1$ by design.

Our regret bound is stated and discussed below. It depends on the minimum and maximum derivatives of the logistic function in the compliance model, which is standard in generalized linear bandit analyses [22, 29]. The *minimum derivative of the mean function* in the neighborhood of ψ_* is

$$\dot{g}_{\min} = \min_{\|a\|_2 \leq 1,\, \|\psi - \psi_*\|_2 \leq 1} \dot{g}(a; \psi) ,$$

where $\dot{g}(a; \psi)$ denotes the derivative of $\mu(a^\top \psi)$ with respect to ψ. The *maximum derivative* is

$$\dot{g}_{\max} = \max_{\|a\|_2 \leq 1,\, \psi \in \Psi} \dot{g}(a; \psi) .$$

The mean function in the logistic model is a sigmoid. Therefore, its maximum derivative is $\dot{g}_{\max} = 1/4$.

Theorem 1. *Let τ be a round such that*

$$\lambda_d(G_{\tau,\psi}) \geq \max \left\{ \frac{d \log(n/d) + 2 \log n}{4 \dot{g}_{\min}^2}, 1 \right\} \tag{5}$$

holds with probability at least $1 - 1/n$. Then the n-round regret of CompUCB *is bounded as*

$$R(n) \leq 2 c_\psi \dot{g}_{\max} \sqrt{\frac{dn}{\log 2} \log\left(1 + \frac{n}{d}\right)} + 2 c_\theta \sqrt{\frac{\lambda^{-1} dn}{\log(1 + 4\lambda^{-1})} \log\left(1 + \frac{4\lambda^{-1} n}{d}\right)} + \tau + 4 .$$

Proof. The claim is proved in Appendix A. □

Our regret bound has two key terms. The c_ψ term is the regret for learning the compliance model. It is $\tilde{O}(d\sqrt{n})$, since $c_\psi = \tilde{O}(\sqrt{d})$, and similar to the regret bounds in logistic bandits [14, 22, 29]. The c_θ term is the regret for learning the reward model. It is $\tilde{O}(d\sqrt{n})$, since $c_\theta = \tilde{O}(\sqrt{d})$, and similar to the regret bounds in linear bandits [1, 10]. Therefore, as expected, our regret bound is the sum of regrets of the two learned models. Finally, we want to comment on the technical assumption in (5). The assumption is borrowed from prior works [22, 29] and can be satisfied as follows. Let ε denote the right-hand side of (5). Then the assumption holds after $\tau = O(\varepsilon d \log n)$ rounds, when the action sets $\mathcal{A}_t$ are sufficiently diverse.

We would like to discuss our proof next. The most novel part is the regret decomposition in the first half, where we decompose the regret into those of logistic and linear models. After that, we apply concentration bounds of Abbasi-Yadkori et al. [1] and Kveton et al. [22], and finish the proof with the elliptical lemma. Our regret decomposition generalizes those in contextual cascading bandits [42] to logistic models. Our attempts to prove a similar bound for CompTS failed. Such bounds are hard to prove in partial observation problems. For instance, Cheung et al. [8] proved a frequentist regret bound for Thompson sampling in a non-contextual cascading bandit with a huge constant of $4\sqrt{\pi} e^{8064}$ (Lemma 4.3 and the last equation in Section 4 therein). Finally, the strength of our proof is modularity, because the concentration arguments can be easily replaced. As an example, we believe that the bound of Kveton et al. [22] could be replaced with that of Faury et al. [13] to improve dependence on the minimum derivative $\dot{g}_{\min}$, which is $\dot{g}_{\min}^{-2}$ now.

5 EXPERIMENTS

We conduct four experiments. In Section 5.2, we experiment with various compliance types. In Section 5.3, we evaluate the robustness of CompTS and CompUCB to model misspecification. In Section 5.4, we study the scalability of our algorithms. Finally, we experiment with a real-world problem in Section 5.5.

Our algorithms, CompTS and CompUCB, are implemented with linear reward and logistic compliance models. In all experiments, the actions are overwritten as follows. If the proposed action A_t does not comply, $C_t = 0$, $\tilde{A}_t$ is a random compliant action. If no action complies, $\tilde{A}_t$ is a random action.

5.1 Baselines

We have three baselines in all experiments: LinTS [4], LinUCB [1], and EnsembleNN [32]. Both LinTS and LinUCB are examples of classic bandit algorithms (Section 2.1). They have the same reward model as CompTS and CompUCB, and illustrate that exploration fails when the compliance of actions is not modeled, even if the reward model is correctly specified.

EnsembleNN is an example of a single model algorithm in Section 2.1. It learns a complex non-linear model of the mean reward of the proposed action, instead of modeling the reward and compliance, as in CompTS and CompUCB. EnsembleNN explores using Thompson sampling, where the true posterior distribution is approximated by an ensemble of neural networks. Each network has 2 fully-connected layers with ReLU activation functions. In each round t, one network from the ensemble is chosen uniformly at random. The action that maximizes the mean reward under the chosen network is taken and all networks in the ensemble are updated using its observed reward. The performance of EnsembleNN is sensitive to its hyper-parameters. Therefore, we tune them. The final tuned values are ensemble size 30, mini-batch window size 32, learning rate 0.2, hidden layer size 100, prior variance 0.01, and perturbation variance 0.05.

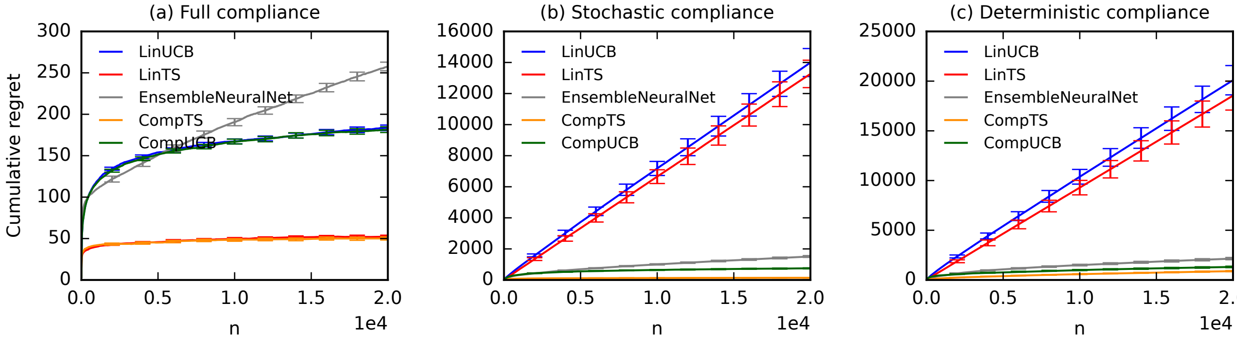

Figure 1: Evaluation of CompTS and CompUCB on three compliance models. From left to right, we experiment with (a) full compliance, (b) stochastic compliance, and (c) deterministic compliance.

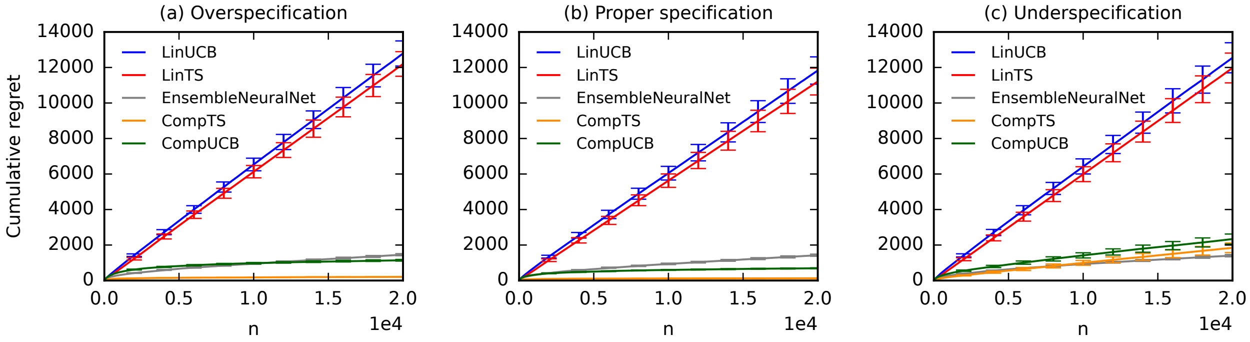

Figure 2: Robustness of CompTS and CompUCB to misspecified compliance models. From left to right, we experiment with (a) feature overspecification, (b) proper feature specification, and (c) feature underspecification.

After the tuning, EnsembleNN performs well in all experiments and shows the versatility of complex models. In comparison, CompTS and CompUCB do not need any tuning and are also less computationally costly. The former is arguably hard in the online setting, since the problem instance is unknown in advance. In terms of computation time, EnsembleNN is 3 times more computationally costly than CompTS and CompUCB in Section 5.2 (750 versus 250 seconds), and 10 times more costly than LinTS and LinUCB (75 seconds). We note that further optimization of all algorithms is possible.

5.2 Compliance Type

We start our experiments with a synthetic problem. The number of features is $d = 10$ and the number of actions is $K = 50$. The reward parameter is sampled i.i.d. from $\mathcal{N}(\mathbf{0}_d, I_d)$. The feature vectors of the actions are sampled uniformly at random from $[-1, 1]^d$. The reward noise is Gaussian $\mathcal{N}(0, 0.5^2)$. All simulations are averaged over 100 independent runs.

Our results are reported in Figure 1. In Figure 1b, we experiment with a stochastic compliance model $C_t \sim \mathrm{Ber}(\mu(A_t^\top \psi_*))$, where the compliance parameter ψ_* is sampled i.i.d. from $\mathcal{N}(\mathbf{0}_d, I_d)$ in each run. This is the same compliance model as in our algorithm design (Section 3) and analysis (Section 4). We observe that learning of the compliance model is very beneficial. Specifically, both LinTS and LinUCB do not model compliance, and have linear regret. Our algorithms also outperform EnsembleNN, which does try to model the structure of our problem.

In Figure 1c, we experiment with deterministic compliance $C_t = \mathbb{1}\{A_{t,1} \geq 0.5\}$, where $A_{t,1}$ is the first entry of the action feature vector A_t. The challenge of this setting is that our model is misspecified, as the actions either always or never comply. Despite this, CompTS and CompUCB learn near-optimal actions, as evidenced by their sublinear regret. In contrast, both LinTS and LinUCB have linear regret. Our algorithms also outperform EnsembleNN.

Finally, in Figure 1a, we experiment with a full compliance model $C_t = 1$. This setting validates that CompTS and CompUCB are implemented correctly. In particular, when all actions comply, CompTS and CompUCB reduce to LinTS and LinUCB, respectively. We also observe that EnsembleNN performs the worst. This is not surprising, since the reward function in this experiment is linear but EnsembleNN tries to approximate it using a 2-layer neural network.

5.3 Model Misspecification

Contextual bandit algorithms are known to be sensitive to model misspecification. In this section, we study this topic. In all experiments, we use the stochastic compliance model from Section 5.2, and modify compliance model features in CompTS and CompUCB as follows. In *feature overspecification*, we add 5 additional compliance features that are sampled uniformly at random from $[-1, 1]$. In this

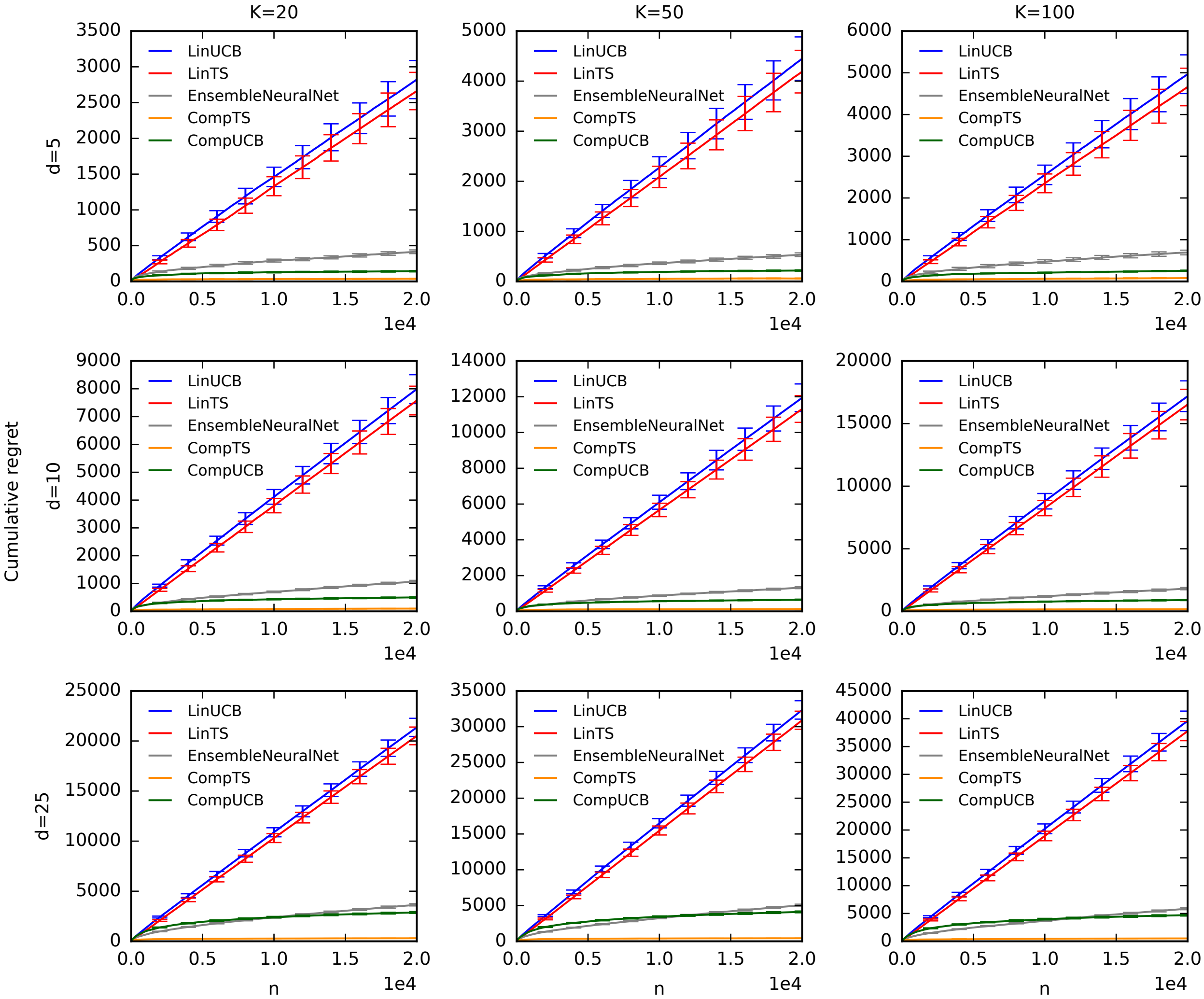

Figure 3: Evaluation of CompTS **and** CompUCB **on** 9 **synthetic problems of different sizes. We vary the number of features, from** $d = 5$ **to** $d = 25$**, and the size of the action set, from** $K = 20$ **to** $K = 100$**.**

case, CompTS and CompUCB have to learn that the additional features are irrelevant, which decreases their statistical efficiency. In *proper feature specification*, the compliance features remain unchanged. Finally, in *feature underspecification*, we randomly remove 2 compliance features out of 10. The challenge of this setting is that the compliance model is misspecified.

Our results are reported in Figure 2. We observe three major trends. First, both CompTS and CompUCB perform well when the compliance features are correctly specified (Figure 2b). Second, when the compliance features are overspecified (Figure 2a), the statistical efficiency of CompTS and CompUCB decreases, which results in a slightly higher regret. Nevertheless, both algorithms still outperform all baselines. Finally, when the compliance features are underspecified, CompTS and CompUCB can have linear regret. In this case, it is most beneficial to learn a more complex model using EnsembleNN. Surprisingly, even learning of an incorrect model of compliance, by CompTS and CompUCB, is more beneficial than not learning it at all, by LinTS and LinUCB.

5.4 Scalability

Now we examine how the regret scales with the number of features d and actions K. All experiments in this section are variants of the stochastic compliance setting in Section 5.2.

Our results are reported in Figure 3. We observe that both CompTS and CompUCB have sublinear regret in all plots, and outperform the classic bandit algorithms, whose regret is always linear. For a fixed d, the regret of CompTS and CompUCB increases slowly with K. This suggests that the proposed algorithms can scale to large action sets. For $d = 25$, CompUCB has a slightly higher regret initially than LinTS and LinUCB. As the number of rounds n increases, CompUCB quickly learns a good policy. The best performing algorithm is CompTS. Its

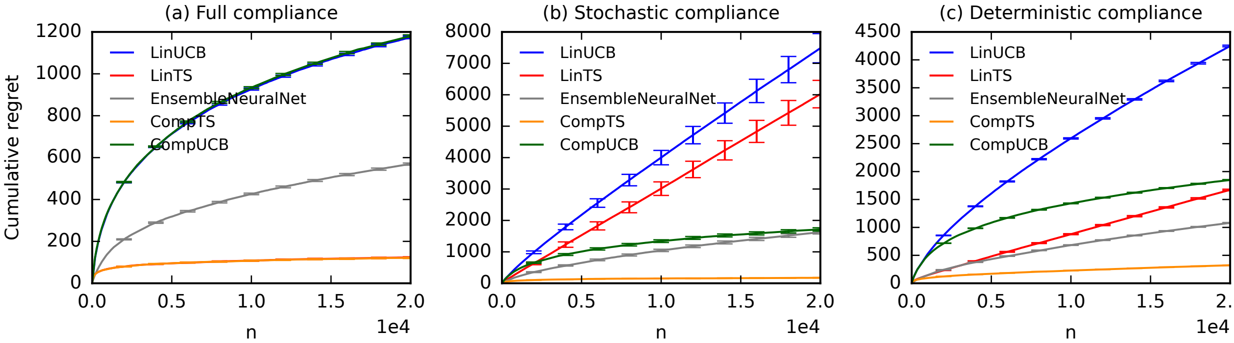

Figure 4: Evaluation of CompTS and CompUCB on three compliance models in the MovieLens dataset. From left to right, we experiment with (a) full compliance, (b) stochastic compliance, and (c) deterministic compliance.

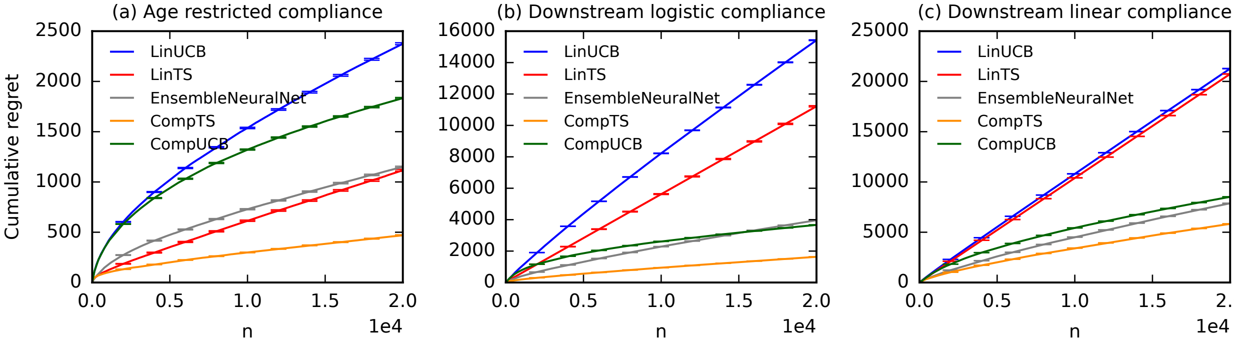

Figure 5: Evaluation of CompTS and CompUCB on three realistic compliance models in the MovieLens dataset. From left to right, we experiment with (a) age restricted compliance, (b) logistic lapse model, and (c) linear activity model.

performance gap over the other algorithms grows with the number of features d.

5.5 MovieLens Experiments

The goal of these experiments is to showcase CompTS and CompUCB beyond synthetic problems. We experiment with the MovieLens 1M dataset [24], with 1 million ratings from 6 040 users for 3 883 movies. The dataset is used as follows. We complete the sparse rating matrix M using alternating least squares [11] with rank $d = 5$. This rank is high enough to yield a low prediction error. The learned factorization is $M = UV^\top$. The i-th row of U, denoted by U_i, is the latent factor representing user i. The j-th row of V, denoted by V_j, is the latent factor representing movie j.

Our results are averaged over 100 random runs. In each run, we simulate interactions of a recommender system with randomly arriving users over n rounds. In interaction $t \in [n]$, we first choose a random user i. Then we select 100 random movies as the action set $\mathcal{A}_t$ in round t. The feature vector of movie j recommended to user i is $\mathrm{vec}(U_i V_j^\top)$, where $\mathrm{vec}(M)$ is a vectorization of matrix M. Since both U_i and V_j are 5-dimensional vectors, $\mathrm{vec}(U_i V_j^\top)$ has 25 dimensions. The mean reward for recommending movie j to user i is $U_i V_j^\top$ and the agent observes it with Gaussian noise $\mathcal{N}(0, 0.5^2)$.

Compliance type. We study the same three compliance types as in Section 5.2. Our results are reported in Figure 4. We observe that all trends are similar to Figure 1. One exception is that CompUCB is less statistically efficient, which is manifested by a higher regret relative to the other algorithms. This is because the number of features d increased. The regret remains sublinear though.

Realistic compliance rules. We experiment with three realistic scenarios. Since the MovieLens dataset does not contain compliance feedback, we generate it based on user and item features. The first non-compliance scenario is *age restricted compliance.* Initially we wanted to block horror and crime movies for young users. Unfortunately, these users comprise only about 3% of the MovieLens dataset. Therefore, we decided to suppress children movies for users above age 18. The other two non-compliance scenarios are motivated by pop-up messages on mobile devices, which are supposed to increase user engagement. In the first scenario, we measure user activity by the number of days since the user left a rating in a 30-day window. We predict the activity using a linear model of user features. A recommendation is suppressed if the user activity is at least 2 days. We call this model *linear compliance.* In the other scenario, we measure user activity using lapse, which takes value 0 if the user is active in the subsequent 30 days, and 1 otherwise. We predict the lapse using a logistic model of user features. A recommendation is suppressed if the lapse is lower than 0.5. We call this model *logistic compliance.* If the suppression was based on user features only, we would either recommend everything or nothing. Therefore, we limit it only to movies j whose first entry of V_j is greater than 0.5, similarly to our experiments in Figures 1c and 4c. The thresholds are chosen so that about 50% of recommendations are suppressed on average.

Our results are reported in Figure 5. In all experiments, CompTS and CompUCB perform well, and CompTS is the best algorithm. Although CompUCB has a higher regret than LinTS in Figure 5a, it is never linear. LinTS and LinUCB have linear regret in Figures 5b and 5c. Overall, we observe that CompTS and CompUCB perform robustly in all tested compliance models.

6 CONCLUSIONS

We study non-compliant bandits, where an agent takes actions that may be overwritten by a downstream algorithm. We propose UCB and Thompson sampling algorithms for this setting, which model the reward and compliance separately. The algorithms are contextual and efficient. We prove a $\tilde{O}(d\sqrt{n})$ regret bound for CompUCB with d features over n rounds. Our empirical results show that CompTS and CompUCB consistently outperform their classic counterparts, LinTS and LinUCB, in both synthetic and real-world problems. In all but one experiment, CompTS outperforms EnsembleNN, which learns an expressive neural network reward model, instead of modeling the structure of our problem. This is despite the fact that EnsembleNN was tuned, which would be impossible in the true online setting.

Non-compliance is understudied in bandits and we hope that this paper will encourage more work on this important practical topic. Our work can be readily extended in four directions. First, we make an assumption that the rewards of taken actions are always observed. Our algorithm design and analysis can be generalized to a slightly more general setting, where the reward is only observed when the proposed action complies. Second, while we show empirically that CompTS performs well, we do not bound its regret. We discuss technical challenges that prevented the analysis after Theorem 1. Third, similarly to classic bandit algorithms, model misspecification can be an issue, and additional work is needed to improve the robustness of CompTS and CompUCB. Finally, our work can be extended to other popular bandit models for recommender systems, such as combinatorial semi-bandits [7, 15, 21], online learning to rank [9, 18, 20, 26, 34], and collaborative filtering and low-rank bandits [19, 30, 37, 41].

A PROOF OF THEOREM 1

To simplify notation, we define $r(a) = g(a; \psi_*) f(a; \theta_*)$. We start with trivial upper bounds on the regret in the first τ rounds and the n-round regret due to (5) not holding in round τ,

$$R(n) \leq \mathbb{E}\left[\sum_{t=\tau}^{n} r(A_{t,*}) - r(A_t)\right] + \tau + 1\,.$$

If (5) holds in round τ, we have $\mathbb{P}\left(\|\hat{\psi}_t - \psi_*\|_2 > 1\right) \leq 1/n$ for any round $t \geq \tau$ [22]. Therefore, we can further bound the regret as

$$R(n) \leq \mathbb{E}\left[\sum_{t=\tau}^{n} \mathbb{1}\{E_t\}\left(r(A_{t,*}) - r(A_t)\right)\right] + \tau + 2\,,$$

where $E_t = \left\{\|\hat{\psi}_t - \psi_*\|_2 \leq 1\right\}$. To simplify notation, we do not write $\mathbb{1}\{E_t\}$ in the rest of the analysis.

Let H_t be the history of all interactions of the agent up to round t. Let $U_{t,\theta}(a)$ and $L_{t,\theta}(a)$ be high-probability upper and lower confidence bounds, respectively, on the mean reward of action a in round t. Let $U_{t,\psi}(a)$ and $L_{t,\psi}(a)$ be the corresponding quantities for the compliance probability of action a in round t. Specifically, let

$$\begin{aligned} L_{t,\theta}(a) \leq f(a; \theta_*) \leq U_{t,\theta}(a)\,, \\ L_{t,\psi}(a) \leq g(a; \psi_*) \leq U_{t,\psi}(a)\,, \end{aligned}$$

hold jointly over all actions a and rounds t with probability at least $1 - 2\delta$. Let

$$\begin{aligned} W_{t,\theta}(a) = U_{t,\theta}(a) - L_{t,\theta}(a)\,, \\ W_{t,\psi}(a) = U_{t,\psi}(a) - L_{t,\psi}(a)\,. \end{aligned}$$

Now we introduce the upper confidence bounds and get

$$\begin{aligned} & r(A_{t,*}) - r(A_t) \qquad (6) \\ & \quad = g(A_{t,*}; \psi_*) f(A_{t,*}; \theta_*) - g(A_t; \psi_*) f(A_t; \theta_*) \\ & \quad \leq g(A_{t,*}; \psi_*) f(A_{t,*}; \theta_*) - U_{t,\psi}(A_{t,*}) U_{t,\theta}(A_{t,*}) + \\ & \qquad U_{t,\psi}(A_t) U_{t,\theta}(A_t) - g(A_t; \psi_*) f(A_t; \theta_*)\,. \end{aligned}$$

The inequality $U_{t,\psi}(A_t)U_{t,\theta}(A_t) \geq U_{t,\psi}(A_{t,*})U_{t,\theta}(A_{t,*})$ holds by the design of CompUCB. The first difference in (6) is at most zero with probability at least $1 - 2\delta$. The second difference in (6) can be decomposed as

$$\begin{aligned} & U_{t,\psi}(A_t) U_{t,\theta}(A_t) - g(A_t; \psi_*) f(A_t; \theta_*) \\ & \quad = U_{t,\theta}(A_t)[U_{t,\psi}(A_t) - g(A_t; \psi_*)] + \\ & \qquad g(A_t; \psi_*)[U_{t,\theta}(A_t) - f(A_t; \theta_*)]\,. \end{aligned}$$

Now we use that $U_{t,\theta}(a) \in [0, 1]$, apply lower confidence bounds to g and f, and get

$$\begin{aligned} & U_{t,\psi}(A_t) U_{t,\theta}(A_t) - g(A_t; \psi_*) f(A_t; \theta_*) \\ & \quad \leq W_{t,\psi}(A_t) + g(A_t; \psi_*) W_{t,\theta}(A_t)\,. \end{aligned}$$

Since $g(A_t; \psi_*)$ and C_t can be exchanged in expectation,

$$\begin{aligned} & \mathbb{E}\left[U_{t,\psi}(A_t) U_{t,\theta}(A_t) - g(A_t; \psi_*) f(A_t; \theta_*)\right] \\ & \quad \leq \mathbb{E}\left[W_{t,\psi}(A_t) + C_t W_{t,\theta}(A_t)\right]\,. \end{aligned}$$

Finally, we collect all inequalities thus far and get

$$R(n) \leq \mathbb{E}\left[\sum_{t=\tau}^{n} W_{t,\psi}(A_t) + C_t W_{t,\theta}(A_t)\right] + \tau + 2 + 2\delta n\,.$$

We instantiate the upper and lower confidence bounds next.

Reward model. The reward model is linear with parameter θ_*. Thus, by Theorem 2 of Abbasi-Yadkori et al. [1], θ_* lies in set

$$I_{t,\theta} = \left\{\theta \in \Theta : \|\theta - \hat{\theta}_t\|_{\hat{\Sigma}_{t,\theta}^{-1}} \leq \sigma\sqrt{d \log\left(\frac{1 + n/\lambda}{\delta}\right)} + \lambda^{\frac{1}{2}} S\right\}$$

in all rounds $t \in [n]$ jointly with probability at least $1 - \delta$, where $\|\theta_*\|_2 \leq S$. In our case, $\sigma = 1/2$, and the definition of $I_{t,\theta}$ leads to an upper confidence bound

$$\begin{aligned}a^\top \theta_* &= a^\top \hat{\theta}_t + a^\top (\theta_* - \hat{\theta}_t) \\ &\leq a^\top \hat{\theta}_t + \|\theta_* - \hat{\theta}_t\|_{\hat{\Sigma}^{-1}_{t,\theta}} \|a\|_{\hat{\Sigma}_{t,\theta}} \\ &\leq a^\top \hat{\theta}_t + \underbrace{\left(\frac{1}{2}\sqrt{d \log\left(\frac{1 + n/\lambda}{\delta}\right)} + \lambda^{\frac{1}{2}} S\right)}_{c_\theta} \|a\|_{\hat{\Sigma}_{t,\theta}} \\ &= U_{t,\theta}(a)\,.\end{aligned}$$

Similarly, $L_{t,\theta}(a) = a^\top \hat{\theta}_t - c_\theta \|a\|_{\hat{\Sigma}_{t,\theta}}$ is a lower confidence bound and thus $W_{t,\theta}(a) \leq 2c_\theta \|a\|_{\hat{\Sigma}_{t,\theta}}$. By the Cauchy-Schwarz inequality and the concavity of the square root,

$$\begin{aligned}\mathbb{E}\left[\sum_{t=\tau}^{n} C_t W_{t,\theta}(A_t)\right] &\leq 2c_\theta \sqrt{n\mathbb{E}\left[\sum_{t=\tau}^{n} C_t \|A_t\|^2_{\hat{\Sigma}_{t,\theta}}\right]} \\ &\leq 2c_\theta \sqrt{n\mathbb{E}\left[\sum_{t=\tau}^{n} \|\tilde{A}_t\|^2_{\hat{\Sigma}_{t,\theta}}\right]}\,.\end{aligned}$$

The last step follows from $A_t = \tilde{A}_t$ when $C_t = 1$ and that the norms are non-negative. Finally, we apply the elliptical lemma (Lemma 2 in Appendix B) for $\Sigma_0 = \lambda^{-1} I_d$, $G_t = G_{t,\theta}$, and $\sigma = 1/2$,

$$\sum_{t=\tau}^{n} C_t \|A_t\|^2_{\hat{\Sigma}_{t,\theta}} \leq \frac{\lambda^{-1} d}{\log(1 + 4\lambda^{-1})} \log\left(1 + \frac{4\lambda^{-1} n}{d}\right)\,.$$

Compliance model. The compliance model is logistic with parameter ψ_*. Therefore, by Lemmas 7 and 8 in Kveton et al. [22], ψ_* lies in set

$$I_{t,\psi} = \left\{\psi \in \Psi : \|\psi - \hat{\psi}_t\|_{\hat{\Sigma}^{-1}_{t,\psi}} \leq \frac{1}{2\dot{g}^2_{\min}} \sqrt{d \log\left(\frac{n}{d\delta}\right)}\right\}$$

in all rounds $t \geq \tau$ jointly with probability at least $1 - \delta$, under the assumption that $\|\hat{\psi}_t - \psi_*\|_2 \leq 1$. The definition of $I_{t,\psi}$ leads to an upper confidence bound

$$\begin{aligned}g(a; \psi_*) &= \mu(a^\top \hat{\psi}_t + a^\top (\psi_* - \hat{\psi}_t)) \\ &\leq \mu(a^\top \hat{\psi}_t + \|\psi_* - \hat{\psi}_t\|_{\hat{\Sigma}^{-1}_{t,\psi}} \|a\|_{\hat{\Sigma}_{t,\psi}}) \\ &\leq \mu\left(a^\top \hat{\psi}_t + \underbrace{\frac{1}{2\dot{g}^2_{\min}} \sqrt{d \log\left(\frac{n}{d\delta}\right)}}_{c_\psi} \|a\|_{\hat{\Sigma}_{t,\psi}}\right) \\ &= U_{t,\psi}(a)\,.\end{aligned}$$

Analogously, $L_{t,\psi}(a) = \mu(a^\top \hat{\psi}_t - c_\psi \|a\|_{\hat{\Sigma}_{t,\psi}})$ is a lower confidence bound and thus $W_{t,\psi}(a) \leq 2c_\psi \dot{g}_{\max} \|a\|_{\hat{\Sigma}_{t,\psi}}$, where $\dot{g}_{\max}$ is the Lipschitz factor of μ. By the Cauchy-Schwarz inequality and the concavity of the square root,

$$\mathbb{E}\left[\sum_{t=\tau}^{n} W_{t,\psi}(A_t)\right] \leq c_\psi \dot{g}_{\max} \sqrt{n\mathbb{E}\left[\sum_{t=\tau}^{n} \|A_t\|^2_{\hat{\Sigma}_{t,\psi}}\right]}\,.$$

Finally, we apply the elliptical lemma (Lemma 2 in Appendix B) for $\Sigma_0 = \hat{\Sigma}_{\tau,\psi}$ and $G_t = G_{t,\psi}$, which implies that $\sigma = 1$. From the definition of τ, $\lambda_1(\Sigma_0) \leq 1$, and thus

$$\sum_{t=\tau}^{n} \|A_t\|^2_{\hat{\Sigma}_{t,\psi}} \leq \frac{d}{\log 2} \log\left(1 + \frac{n}{d}\right)\,.$$

To complete the proof, we set $\delta = 1/n$.

B TECHNICAL LEMMAS

Lemma 2. *Let* $G_t = \sigma^{-2} \sum_{\ell=1}^{t-1} A_\ell A_\ell^\top$ *and* $\hat{\Sigma}_t = (\Sigma_0^{-1} + G_t)^{-1}$. *Then for any PSD matrix* Σ_0, $(A_t)_{t\in[n]}$ *of at most unit length, and* $\sigma > 0$,

$$\sum_{t=1}^{n} \|A_t\|^2_{\hat{\Sigma}_t} \leq dc \log\left(1 + \frac{\lambda_1(\Sigma_0) n}{\sigma^2 d}\right), \quad c = \frac{\lambda_1(\Sigma_0)}{\log(1 + \sigma^{-2}\lambda_1(\Sigma_0))}\,.$$

Proof. Fix round t and note that

$$\begin{aligned}\|A_t\|^2_{\hat{\Sigma}_t} &= \sigma^2 \frac{A_t^\top \hat{\Sigma}_t A_t}{\sigma^2} \leq c \log(1 + \sigma^{-2} A_t^\top \hat{\Sigma}_t A_t) \qquad (7) \\ &= c \log\det(I_d + \sigma^{-2} \hat{\Sigma}_t^{\frac{1}{2}} A_t A_t^\top \hat{\Sigma}_t^{\frac{1}{2}})\end{aligned}$$

holds for c defined in the claim. The inequality is proved as follows. For any $x \in (0, u]$,

$$\begin{aligned}x &= \frac{x}{\log(1+x)} \log(1+x) \leq \left(\max_{x\in[0,u]} \frac{x}{\log(1+x)}\right) \log(1+x) \\ &= \frac{u}{\log(1+u)} \log(1+x)\,.\end{aligned}$$

Then we set $x = \sigma^{-2} A_t^\top \hat{\Sigma}_t A_t$ and use Weyl's inequalities to derive

$$\begin{aligned}A_t^\top \hat{\Sigma}_t A_t &\leq \lambda_1(\hat{\Sigma}_t) = \lambda_1((\Sigma_0^{-1} + G_t)^{-1}) = \lambda_d^{-1}(\Sigma_0^{-1} + G_t) \\ &\leq \lambda_d^{-1}(\Sigma_0^{-1}) = \lambda_1(\Sigma_0)\,.\end{aligned}$$

The next step is an upper bound on the logarithmic term in (7) that can be rewritten as

$$\begin{aligned}&\log\det(I_d + \sigma^{-2} \hat{\Sigma}_t^{\frac{1}{2}} A_t A_t^\top \hat{\Sigma}_t^{\frac{1}{2}}) \\ &\quad = \log\det(\hat{\Sigma}_t^{-1} + \sigma^{-2} A_t A_t^\top) - \log\det(\hat{\Sigma}_t^{-1})\,.\end{aligned}$$

Because of that, when we sum over all rounds, we get telescoping and the total contribution of all terms is at most

$$\begin{aligned}&\sum_{t=1}^{n} \log\det(I_d + \sigma^{-2} \hat{\Sigma}_t^{\frac{1}{2}} A_t A_t^\top \hat{\Sigma}_t^{\frac{1}{2}}) \\ &\quad = \log\det(\hat{\Sigma}_{n+1}^{-1}) - \log\det(\hat{\Sigma}_1^{-1}) = \log\det(\Sigma_0^{\frac{1}{2}} \hat{\Sigma}_{n+1}^{-1} \Sigma_0^{\frac{1}{2}}) \\ &\quad \leq d \log\left(\frac{1}{d} \operatorname{tr}(\Sigma_0^{\frac{1}{2}} \hat{\Sigma}_{n+1}^{-1} \Sigma_0^{\frac{1}{2}})\right) \\ &\quad = d \log\left(1 + \frac{1}{\sigma^2 d} \sum_{t=1}^{n} \operatorname{tr}(\Sigma_0^{\frac{1}{2}} A_t A_t^\top \Sigma_0^{\frac{1}{2}})\right) \\ &\quad = d \log\left(1 + \frac{1}{\sigma^2 d} \sum_{t=1}^{n} A_t^\top \Sigma_0 A_t\right) \leq d \log\left(1 + \frac{\lambda_1(\Sigma_0) n}{\sigma^2 d}\right)\,.\end{aligned}$$

This completes the proof. □

REFERENCES

[1] Yasin Abbasi-Yadkori, David Pal, and Csaba Szepesvari. 2011. Improved Algorithms for Linear Stochastic Bandits. In *Advances in Neural Information Processing Systems 24*. 2312–2320.

[2] Marc Abeille and Alessandro Lazaric. 2017. Linear Thompson Sampling Revisited. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*.

[3] Shipra Agrawal and Navin Goyal. 2012. Analysis of Thompson Sampling for the Multi-Armed Bandit Problem. In *Proceeding of the 25th Annual Conference on Learning Theory*. 39.1–39.26.

[4] Shipra Agrawal and Navin Goyal. 2013. Thompson Sampling for Contextual Bandits with Linear Payoffs. In *Proceedings of the 30th International Conference on Machine Learning*. 127–135.

[5] Peter Auer, Nicolo Cesa-Bianchi, and Paul Fischer. 2002. Finite-time Analysis of the Multiarmed Bandit Problem. *Machine Learning* 47 (2002), 235–256.

[6] Ruey-Cheng Chen, Luke Gallagher, Roi Blanco, and J. Shane Culpepper. 2017. Efficient Cost-Aware Cascade Ranking in Multi-Stage Retrieval. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 445–454.

[7] Wei Chen, Yajun Wang, and Yang Yuan. 2013. Combinatorial Multi-Armed Bandit: General Framework, Results and Applications. In *Proceedings of the 30th International Conference on Machine Learning*. 151–159.

[8] Wang Chi Cheung, Vincent Tan, and Zixin Zhong. 2019. A Thompson Sampling Algorithm for Cascading Bandits. In *Proceedings of the 22nd International Conference on Artificial Intelligence and Statistics*. 438–447.

[9] Richard Combes, Stefan Magureanu, Alexandre Proutiere, and Cyrille Laroche. 2015. Learning to Rank: Regret Lower Bounds and Efficient Algorithms. In *Proceedings of the 2015 ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems*.

[10] Varsha Dani, Thomas Hayes, and Sham Kakade. 2008. Stochastic Linear Optimization under Bandit Feedback. In *Proceedings of the 21st Annual Conference on Learning Theory*. 355–366.

[11] Mark Davenport and Justin Romberg. 2016. An Overview of Low-Rank Matrix Recovery From Incomplete Observations. *IEEE Journal of Selected Topics in Signal Processing* 10, 4 (2016), 608–622.

[12] Qin Ding, Cho-Jui Hsieh, and James Sharpnack. 2021. An Efficient Algorithm For Generalized Linear Bandit: Online Stochastic Gradient Descent and Thompson Sampling. In *Proceedings of the 24th International Conference on Artificial Intelligence and Statistics*.

[13] Louis Faury, Marc Abeille, Clement Calauzenes, and Olivier Fercoq. 2020. Improved Optimistic Algorithms for Logistic Bandits. In *Proceedings of the 37th International Conference on Machine Learning*.

[14] Sarah Filippi, Olivier Cappe, Aurelien Garivier, and Csaba Szepesvari. 2010. Parametric Bandits: The Generalized Linear Case. In *Advances in Neural Information Processing Systems 23*. 586–594.

[15] Yi Gai, Bhaskar Krishnamachari, and Rahul Jain. 2012. Combinatorial Network Optimization with Unknown Variables: Multi-Armed Bandits with Linear Rewards and Individual Observations. *IEEE/ACM Transactions on Networking* 20, 5 (2012), 1466–1478.

[16] Kwang-Sung Jun, Aniruddha Bhargava, Robert Nowak, and Rebecca Willett. 2017. Scalable Generalized Linear Bandits: Online Computation and Hashing. In *Advances in Neural Information Processing Systems 30*. 98–108.

[17] Nathan Kallus. 2018. Instrument-Armed Bandits. In *Proceedings of the 29th International Conference on Algorithmic Learning Theory*. 529–546.

[18] Sumeet Katariya, Branislav Kveton, Csaba Szepesvari, and Zheng Wen. 2016. DCM Bandits: Learning to Rank with Multiple Clicks. In *Proceedings of the 33rd International Conference on Machine Learning*. 1215–1224.

[19] Jaya Kawale, Hung Bui, Branislav Kveton, Long Tran-Thanh, and Sanjay Chawla. 2015. Efficient Thompson Sampling for Online Matrix-Factorization Recommendation. In *Advances in Neural Information Processing Systems 28*. 1297–1305.

[20] Branislav Kveton, Csaba Szepesvari, Zheng Wen, and Azin Ashkan. 2015. Cascading Bandits: Learning to Rank in the Cascade Model. In *Proceedings of the 32nd International Conference on Machine Learning*.

[21] Branislav Kveton, Zheng Wen, Azin Ashkan, and Csaba Szepesvari. 2015. Tight Regret Bounds for Stochastic Combinatorial Semi-Bandits. In *Proceedings of the 18th International Conference on Artificial Intelligence and Statistics*.

[22] Branislav Kveton, Manzil Zaheer, Csaba Szepesvari, Lihong Li, Mohammad Ghavamzadeh, and Craig Boutilier. 2020. Randomized Exploration in Generalized Linear Bandits. In *Proceedings of the 23rd International Conference on Artificial Intelligence and Statistics*.

[23] Tze Leung Lai and Herbert Robbins. 1985. Asymptotically Efficient Adaptive Allocation Rules. *Advances in Applied Mathematics* 6, 1 (1985), 4–22.

[24] Shyong Lam and Jon Herlocker. 2016. MovieLens Dataset. http://grouplens.org/datasets/movielens/.

[25] John Langford and Tong Zhang. 2008. The Epoch-Greedy Algorithm for Contextual Multi-Armed Bandits. In *Advances in Neural Information Processing Systems 20*. 817–824.

[26] Tor Lattimore, Branislav Kveton, Shuai Li, and Csaba Szepesvari. 2018. TopRank: A Practical Algorithm for Online Stochastic Ranking. In *Advances in Neural Information Processing Systems 31*. 3949–3958.

[27] Tor Lattimore and Csaba Szepesvari. 2019. *Bandit Algorithms*. Cambridge University Press.

[28] Lihong Li, Wei Chu, John Langford, and Robert Schapire. 2010. A Contextual-Bandit Approach to Personalized News Article Recommendation. In *Proceedings of the 19th International Conference on World Wide Web*.

[29] Lihong Li, Yu Lu, and Dengyong Zhou. 2017. Provably Optimal Algorithms for Generalized Linear Contextual Bandits. In *Proceedings of the 34th International Conference on Machine Learning*. 2071–2080.

[30] Shuai Li, Alexandros Karatzoglou, and Claudio Gentile. 2016. Collaborative Filtering Bandits. In *Proceedings of the 39th Annual International ACM SIGIR Conference*.

[31] Yi Liu and Lihong Li. 2021. A Map of Bandits for E-Commerce. In *KDD 2021 Workshop on Multi-Armed Bandits and Reinforcement Learning*.

[32] Xiuyuan Lu and Benjamin Van Roy. 2017. Ensemble Sampling. In *Advances in Neural Information Processing Systems 30*. 3258–3266.

[33] Nicolas Della Penna, Mark Reid, and David Balduzzi. 2016. Compliance-Aware Bandits. *CoRR* abs/1602.02852 (2016). http://arxiv.org/abs/1602.02852

[34] Filip Radlinski, Robert Kleinberg, and Thorsten Joachims. 2008. Learning Diverse Rankings with Multi-Armed Bandits. In *Proceedings of the 25th International Conference on Machine Learning*. 784–791.

[35] Daniel Russo, Benjamin Van Roy, Abbas Kazerouni, Ian Osband, and Zheng Wen. 2018. A Tutorial on Thompson Sampling. *Foundations and Trends in Machine Learning* 11, 1 (2018), 1–96.

[36] Shubhra Kanti Karmaker Santu, Parikshit Sondhi, and ChengXiang Zhai. 2017. On Application of Learning to Rank for E-Commerce Search. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 475–484.

[37] Rajat Sen, Karthikeyan Shanmugam, Murat Kocaoglu, Alex Dimakis, and Sanjay Shakkottai. 2017. Contextual Bandits with Latent Confounders: An NMF Approach. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*.

[38] Andrew Stirn and Tony Jebara. 2018. Thompson Sampling for Noncompliant Bandits. *CoRR* abs/1812.00856 (2018). http://arxiv.org/abs/1812.00856

[39] William R. Thompson. 1933. On the Likelihood that One Unknown Probability Exceeds Another in View of the Evidence of Two Samples. *Biometrika* 25, 3-4 (1933), 285–294.

[40] R. Wolke and H. Schwetlick. 1988. Iteratively Reweighted Least Squares: Algorithms, Convergence Analysis, and Numerical Comparisons. *SIAM J. Sci. Statist. Comput.* 9, 5 (1988), 907–921.

[41] Xiaoxue Zhao, Weinan Zhang, and Jun Wang. 2013. Interactive Collaborative Filtering. In *Proceedings of the 22nd ACM International Conference on Information and Knowledge Management*. 1411–1420.

[42] Shi Zong, Hao Ni, Kenny Sung, Nan Rosemary Ke, Zheng Wen, and Branislav Kveton. 2016. Cascading Bandits for Large-Scale Recommendation Problems. In *Proceedings of the 32nd Conference on Uncertainty in Artificial Intelligence*.