Contextual Multi-armed Bandit Algorithm for Semiparametric Reward Model

Gi-Soo Kim [1] **Myunghee Cho Paik** [1]

Abstract

Contextual multi-armed bandit (MAB) algorithms have been shown promising for maximizing cumulative rewards in sequential decision tasks such as news article recommendation systems, web page ad placement algorithms, and mobile health. However, most of the proposed contextual MAB algorithms assume linear relationships between the reward and the context of the action. This paper proposes a new contextual MAB algorithm for a relaxed, semiparametric reward model that supports nonstationarity. The proposed method is less restrictive, easier to implement and faster than two alternative algorithms that consider the same model, while achieving a tight regret upper bound. We prove that the high-probability upper bound of the regret incurred by the proposed algorithm has the same order as the Thompson sampling algorithm for linear reward models. The proposed and existing algorithms are evaluated via simulation and also applied to Yahoo! news article recommendation log data.

1. Introduction

The multi-armed bandit (MAB) problem (Robbins, 1952) formulates the sequential decision problem in which a learner must choose an action among several actions given by the environment at each step so as to maximize the cumulative rewards. The actions are often described as the arms of a bandit slot machine with multiple arms. By choosing an action or pulling an arm, the learner receives possibly different rewards. By repeating the process of pulling arms and receiving rewards, the learner accumulates information about the reward compensation mechanism, learns from it, and chooses the arm close to optimal as time elapses. Application areas include the mobile healthcare system (Tewari and Murphy, 2017), web page ad placement algorithms (Langford et al., 2008), news article placement algorithms (Li et al., 2010), revenue management (Ferreira et al., 2018), marketing (Schwartz et al., 2017), and recommendation systems (Kawale et al., 2015).

For example, the Yahoo! web system uses a news article recommendation algorithm to select one article among a large pool of available articles and displays it on the Featured tab of the web page every time a user visits. The user clicks the article if he or she is interested in the contents. The goal of the algorithm is to maximize the cumulative number of user clicks. After each visit, the algorithm reinforces its article selection strategy based on the past user click feedback. In this setting, available articles correspond to different actions and the user click corresponds to a reward. The challenging part of the MAB problem is that the reward of the action that the learner has not previously chosen is forever unknown, i.e., whether the user would have clicked or not remains missing for the article that is not chosen. Therefore, the learner should balance between "exploitation", selecting the best action based on information accumulated so far, and "exploration", choosing an action that will assist in future choices, although it does not seem to be the best option at the moment.

The MAB problem was first theoretically analyzed by Lai and Robbins (1985). The algorithms widely used in mobile healthcare systems or news article placement algorithms are of a more extended form, called contextual MAB algorithms. A contextual MAB algorithm enables at each selection step the use of side information, called context, about each action given in the form of finite-dimensional covariates. For example, in the news article recommendation, information on the visiting user as well as the articles are given in the form of a context vector. In 2010, the Yahoo! team (Li et al., 2010) proposed a contextual MAB algorithm that achieved a 12.5% click lift compared to a context-free MAB algorithm. Nonetheless, the method of Li et al. (2010) and other existing algorithms rely on rather strong assumptions on the distribution of rewards. In particular, most of the existing algorithms assume that the expectation of the reward of a particular action has a time-invariant linear relationship with the context vector. This assumption can be restrictive in real world settings where the rewards typically vary with time.

In this paper, we propose a novel contextual MAB algorithm

[1]Department of Statistics, Seoul National University, Seoul, Korea. Correspondence to: Myunghee Cho Paik <myunghee-chopaik@snu.ac.kr>.

Proceedings of the 36^{th} International Conference on Machine Learning, Long Beach, California, PMLR 97, 2019.

which works well under a relaxed assumption on the distribution of rewards. The relaxed nature of the assumption involves nonstationarity of the reward via an additive intercept term to the original time-invariant linear term. This intercept term changes with time but does not depend on the action. We propose a consistent estimation of the regression parameter in the linear term by centering the context vectors with weights. Using new martingale inequalities, we prove that the high probability upper bound of the total regret incurred by the proposed algorithm has the same order as the regret bound achieved by the Thompson sampling algorithm which is developed under a more restrictive linear assumption.

Greenewald et al. (2017) and Krishnamurthy et al. (2018) suggested algorithms under the same nonstationary assumption we considered. The performance of the method by Greenewald et al. (2017) is guaranteed under restrictive conditions on the action choice probabilities. The method by Krishnamurthy et al. (2018) takes an action-elimination approach which is computationally heavy as it requires $O(N^2)$ computations at each iteration where N denotes the number of arms. Moreover in Krishnamurthy et al. (2018), the action selection distribution is not given explicitly when $N > 2$. Our method improves on these previous results in that it does not restrict action choice probabilities, requires $O(N)$ computations, and explicitly provides the action selection distribution for every N. Furthermore, the proposed estimator for the regression parameter achieves the same convergence rate as the estimator for linear reward models.

As a summary, our main contributions are:

- We propose a new MAB algorithm for the nonstationary semiparametric reward model. The proposed method is less restrictive, easier to implement and computationally faster than previous works.
- We prove that the high-probability upper bound of the regret for the proposed method is of the same order as the Thompson Sampling algorithm for linear reward models.
- We propose a new estimator for the regression parameter without requiring an extra tuning parameter and prove that it converges to the true parameter faster than existing estimators.
- Simulation studies show that in most cases, the cumulative reward of the proposed method increases faster than existing methods which assume the same nonstationary reward model. Application to Yahoo! news article recommendation log data shows that the proposed method increases the user click rate compared to the algorithms that assume a stationary reward model.

2. Preliminaries

In this section, we describe the problem settings and notations. As a preliminary, we also present a review of contextual bandit methods for the comparison purpose with the proposed method given in Section 4.

In the MAB setting, the learner is repeatedly faced with N alternative actions where at time t, the i-th arm ($i = 1, \cdots, N$) yields a random reward $r_i(t)$ with unknown mean $\theta_i(t)$. In the contextual MAB problem, we assume that there is a finite-dimensional context vector $b_i(t) \in \mathbb{R}^d$ associated with each arm i at time t and that the mean of $r_i(t)$ depends on $b_i(t)$, i.e., $\theta_i(t) = \theta_t(b_i(t))$, where $\theta_t(\cdot)$ is an arbitrary function. Among the N arms, the learner pulls one arm $a(t)$, and observes reward $r_{a(t)}(t)$. The optimal arm at time t is $a^*(t) := \underset{1 \le i \le N}{\operatorname{argmax}}\{\theta_t(b_i(t))\}$. Let $regret(t)$ be the difference between the expected reward of the optimal arm and the expected reward of the arm chosen by the learner at time t, i.e.,

$$\begin{aligned} regret(t) &= \mathbb{E}\big(r_{a^*(t)}(t) - r_{a(t)}(t) \,\big|\, \{b_i(t)\}_{i=1}^N, a(t)\,\big) \\ &= \theta_t(b_{a^*(t)}(t)) - \theta_t(b_{a(t)}(t)). \end{aligned}$$

Then, the goal of the learner is to minimize the sum of regrets over T steps, $R(T) := \sum_{t=1}^T regret(t)$.

Linear contextual MAB problems specifically assume that $\theta_t(b_i(t))$ is linear in $b_i(t)$,

$$\theta_t(b_i(t)) = b_i(t)^T \mu, \quad i = 1, \cdots, N, \tag{1}$$

where $\mu \in \mathbb{R}^d$ is unknown. For the linear contextual MAB problem, Dani et al. (2008) proved a lower bound of order $\Omega(d\sqrt{T})$ for the regret $R(T)$ when N is allowed to be infinite. When N is finite and $d^2 \le T$, Chu et al. (2011) showed a lower bound of $\Omega(\sqrt{dT})$.

Auer (2002), Li et al. (2010) and Chu et al. (2011) proposed an upper confidence bound (UCB) algorithm for the linear contextual MAB problem. The algorithm selects the arm which has the highest UCB of the reward. Since the UCB reflects the current estimate of the reward as well as its uncertainty, the algorithm naturally balances between exploitation and exploration. The success of the UCB algorithm hinges on a valid upper confidence bound $U_i(t)$ of the i-th arm's reward, $b_i(t)^T\mu$. Li et al. (2010) and Chu et al. (2011) proposed

$$U_i(t) = b_i(t)^T \hat{\mu}(t) + \alpha s_{t,i},$$

where $\hat{\mu}(t)$ is the regression estimator of μ at time t,

$$\hat{\mu}(t) = B(t)^{-1} \sum_{\tau=1}^{t-1} b_{a(\tau)}(\tau) r_{a(\tau)}(\tau), \tag{2}$$

$B(t) = I_d + \sum_{\tau=1}^{t-1} b_{a(\tau)}(\tau) b_{a(\tau)}(\tau)^T$ and $s_{t,i} = \sqrt{b_i(t)^T B(t)^{-1} b_i(t)}$.

Under additional assumptions that the error $\eta_i(t) := r_i(t) - \mathbb{E}(r_i(t)|b_i(t)) = r_i(t) - b_i(t)^T\mu$ is R-sub-Gaussian for some $R > 0$ and that the L_2-norms of $b_i(t)$ and μ are bounded by 1, Abbasi-Yadkori et al. (2011) proved that if we set $\alpha = R\sqrt{3d\log(T/\delta)} + 1$, $U_i(t)$ is a $(1-\delta)$-probability upper bound of $b_i(t)^T\mu$ for $\forall\delta \in (0,1)$, for all $i = 1, \cdots, N$ and $t = 1, \cdots, T$. Since the errors $\eta_{a(\tau)}(\tau)$'s of the observed rewards are intercorrelated, Abbasi-Yadkori et al. (2011) used a concentration inequality for vector-valued martingales to derive a tight α. Additionally, Abbasi-Yadkori et al. (2011) proved that with probability at least $1-\delta$, the UCB algorithm achieves,

$$R(T) \leq O\big(d\sqrt{T\log(T/\delta)\log(1+T/d)}\big). \tag{3}$$

The bound (3) matches the lower bound $\Omega(d\sqrt{T})$ for infinite N by a factor of $\log(T)$. When N is finite, (3) is slightly higher than the lower bound $\Omega(\sqrt{dT})$ by a factor of $\sqrt{d\log(T)}$.

Thompson sampling (Thompson, 1933) has been widely used as a simple heuristic based on Bayesian ideas. Agrawal and Goyal (2013) was the first to propose and analyze the Thompson sampling (TS) algorithm for linear contextual MABs.

The heuristic of the algorithm is to randomly pull the arm according to the posterior probability that it is the optimal arm. This can be done by sampling $\tilde{\mu}(t)$ from the posterior distribution of μ at time t, and pulling the arm $a(t) = \underset{1\leq i\leq N}{\operatorname{argmax}}\ b_i(t)^T\tilde{\mu}(t)$. The posterior distribution $\mathcal{N}(\hat{\mu}(t), v^2B(t)^{-1})$ with $\hat{\mu}(t)$ defined in (2) is easily derived by assuming a gaussian prior $\mathcal{N}(0_d, v^2I_d)$ on μ for some $v > 0$ and that $r_i(t)$ given μ follows a gaussian distribution $\mathcal{N}(b_i(t)^T\mu, v^2)$.

Agrawal and Goyal (2013) derived the high-probability upper bound of $R(T)$ for the TS algorithm. This bound does not require the Bayesian framework nor the gaussian assumption for the rewards. Under (1) and R-sub-gaussianity of the errors, it can be shown that with probability greater than $1-\delta$,

$$R(T) \leq O\big(d^{\frac{3}{2}}\sqrt{T\log(Td)\log(T/\delta)}(\sqrt{\log(1+T/d)} + \sqrt{\log(1/\delta)}\ )\big). \tag{4}$$

The bound (4) matches the bound (3) by a factor of $\sqrt{d}\sqrt{\log(T)}$, which is the price for randomness. On the other hand, the TS algorithm does not require to compute the $s'_{t,i}s$ for each arm i at every time t, which is computationally advantageous when N is large.

Unlike aforementioned linear contextual MABs, adversarial contextual MABs do not impose a stationary assumption on $\theta_t(\cdot)$. Hence, the distribution of $r_i(t)$ is allowed to change over time, and it can also change adaptively depending on the history. In fact, we assume that an unknown adversary controls the value of $r_i(t)$ in a way that hampers the learner. In this relaxed setting, it is hard to achieve low $regret(t)$ with respect to the best choice $r_{a^*(t)}(t)$. Instead, the learner competes with a predefined, finite set of K policies and the regret is defined with respect to the best policy in that set.

The EXP4.P algorithm proposed by Beygelzimer et al. (2011) achieves $O(\sqrt{TN\log(K/\delta)})$ regret upper bound. However, for the best policy in the predefined set to be close to the optimal policy which chooses $a^*(t)$ for every t, K should be as large as possible, resulting in actual large regret. Therefore, when a simple parametric or semiparametric assumption is not considered so farfetched, algorithms that exploit this structure can have better performance than adversarial MAB algorithms.

3. Semiparametric Contextual MAB

Greenewald et al. (2017) and Krishnamurthy et al. (2018) considered a middle ground between stochastic, stationary, linear contextual MABs and complex adversarial MABs: a semiparametric contextual MAB. In this section, we formally present the semiparametric contextual MAB problem and related works.

3.1. Semiparametric Additive Reward Model

Hereinafter, we define $\mathcal{H}_{t-1}$ as the history until time $t-1$, i.e., $\mathcal{H}_{t-1} = \{a(\tau), r_{a(\tau)}(\tau), \{b_i(\tau)\}_{i=1}^N, \tau = 1, \cdots, t-1\}$, and the filtration $\mathcal{F}_{t-1}$ as the union of $\mathcal{H}_{t-1}$ and the contexts at time t, i.e., $\mathcal{F}_{t-1} = \{\mathcal{H}_{t-1}, \{b_i(t)\}_{i=1}^N\}$ for $t = 1, \cdots, T$. Given $\mathcal{F}_{t-1}$, we assume that the expectation of the reward $r_i(t)$ can be decomposed into a time-invariant, linear component depending on action $(b_i(t)^T\mu)$ and a non-parametric component depending on time and possibly on $\mathcal{F}_{t-1}$, but not on the action $(\nu(t))$:

$$\mathbb{E}(r_i(t)|\mathcal{F}_{t-1}) = \nu(t) + b_i(t)^T\mu. \tag{5}$$

In (5), we do not impose any distributional assumption on $\nu(t)$ except that it is bounded, $|\nu(t)| \leq 1$. If $\nu(t) = 0$, the problem reduces to a linear contextual MAB problem, whereas if $\nu(t)$ depends on the action as well, the reward distribution is completely nonparametric and can be addressed by adversarial MAB algorithms.

In the news article recommendation example, $\nu(t)$ can represent the baseline tendency of the user visiting at time t to click any article in the Featured tab, regardless of the contents of the article. This baseline tendency can change in an unexpected manner, because different users visit at each time and the clicking tendency can change within the same user according to the user's mood or schedule, both

of which cannot be captured as contextual information. It is reasonable to assume that given this baseline tendency, the probability that the user clicks an article is linear with respect to context information of the article and the user.

Under (5), we note that the optimal action $a^*(t)$ at time t does not depend on $\nu(t)$ but only on the value of μ, and the regret does not depend on $\nu(t)$ either:

$$regret(t) = b_{a^*(t)}(t)^T\mu - b_{a(t)}(t)^T\mu.$$

However, $\nu(t)$ confounds the estimation of μ. The nature of the bandit problem renders the distinction of $\nu(t)$ from the linear part especially difficult because only one observation is allowed at each time t.

Besides (5), we make the usual assumption that given $\mathcal{F}_{t-1}$, the error $\eta_i(t) := r_i(t) - \mathbb{E}(r_i(t)|\mathcal{F}_{t-1})$ is R-sub-Gaussian for some $R > 0$, i.e., for every $\lambda \in \mathbb{R}$,

$$\mathbb{E}[\exp(\lambda\eta_i(t))|\mathcal{F}_{t-1}] \leq \exp(\lambda^2R^2/2). \tag{6}$$

Note that this assumption is satisfied whenever $r_i(t) \in [\nu(t) + b_i(t)^T\mu - R, \nu(t) + b_i(t)^T\mu + R]$. Also without loss of generality, we assume

$$||b_i(t)||_2 \leq 1,\ ||\mu||_2 \leq 1,\ |\nu(t)| \leq 1, \tag{7}$$

where $||\cdot||_p$ denotes the L_p-norm.

3.2. Related Work

Greenewald et al. (2017) proposed the action-centered TS algorithm for the new reward model (5). In their settings, they assumed that the first action is the base action, of which the context vector is $b_1(t) = 0_d$ for all t. Hence, the expected reward of the base action is $\nu(t)$, which can vary with time and also in a way that depends on the past. Greenewald et al. (2017) followed the basic framework of the randomized, TS algorithm but in two stages. In the first stage, the learner selects one action among the non-base actions in the same way as in TS algorithm using random $\tilde{\mu}(t)$. Let this action be $\bar{a}(t)$. In the second stage, the learner chooses once again between $\bar{a}(t)$ and the base action using the distribution of $\tilde{\mu}(t)$. This finally chosen action is set as $a(t)$ and only this action is actually taken. In the second stage, the probability of $a(t) = \bar{a}(t)$ is computed using the Gaussian distribution of $\tilde{\mu}(t)$, $\mathbb{P}(a(t) = \bar{a}(t)|\mathcal{F}_{t-1}, \bar{a}(t)) = 1 - \psi\Big(\frac{-b_{\bar{a}(t)}(t)^T\hat{\mu}(t)}{vs_{t,\bar{a}(t)}(t)}\Big)$, where $\psi(\cdot)$ is the CDF of the standard Gaussian distribution.

Instead of choosing $a(t) = \bar{a}(t)$ with this exact probability however, Greenewald et al. (2017) constrained the probability of not choosing the base action to lie in a predefined set $[p_{min}, p_{max}] \subset [0, 1]$. This is to prevent the algorithm from converging to a deterministic policy which can be ineffective in the mobile health setting that the authors considered. Hence, the algorithm selects $a(t) = \bar{a}(t)$ with probability $p_t = \max\Big(p_{min},\ \min\Big(1 - \psi\Big(\frac{-b_{\bar{a}(t)}(t)^T\hat{\mu}(t)}{vs_{t,\bar{a}(t)}(t)}\Big),\ p_{max}\Big)\Big)$. Under this probability constraint, the definition of the optimal policy and $regret(t)$ changes accordingly. Let $\bar{a}^*(t) = \underset{2\leq i\leq N}{\text{argmax}}\ b_i(t)^T\mu$. Thus, $\bar{a}^*(t)$ is the optimal action among the non-base actions. Then the optimal policy chooses the action $a^*(t) = \bar{a}^*(t)$ with probability $\pi^*(t) := p_{max}I(b_{\bar{a}^*(t)}(t)^T\mu > 0) + p_{min}I(b_{\bar{a}^*(t)}(t)^T\mu \leq 0)$ and $a^*(t) = 1$ with probability $1 - \pi^*(t)$.

To consistently estimate μ, Greenewald et al. (2017) defined a pseudo-reward, $\hat{r}_{\bar{a}(t)}(t) = \{I(a(t) = \bar{a}(t)) - p_t\}r_{a(t)}(t)$. An important property of the pseudo-reward is that its conditional expectation does not depend on $\nu(t)$. Greenewald et al. (2017) used this pseudo-reward instead of the actual reward $r_{a(t)}(t)$ for estimating μ. They showed that the high probability upper bound of $R(T)$ for the action-centered TS algorithm matches that of the original TS algorithm for linear reward models, but by a constant factor $M = 1/\{p_{min}(1 - p_{max})\}$. This factor M can be large when we do not want to restrict action selection probabilities, i.e., when we want to set either $p_{min} = 0$ or $p_{max} = 1$.

Krishnamurthy et al. (2018) proposed the BOSE (Bandit Orthogonalized Semiparametric Estimation) algorithm for the semiparametric reward model (5). This algorithm takes an action elimination method adapted from Even-Dar et al. (2006). At each time t, an action i is eliminated if there exists another action j such that $\big(b_j(t) - b_i(t)\big)^T\hat{\mu}(t) > \omega\sqrt{(b_i(t) - b_j(t))^TV_t^{-1}(b_i(t) - b_j(t))}$, where ω is a predefined constant, $\hat{\mu}(t)$ is an estimate of μ, and V_t is a d-dimensional matrix. The algorithm then picks up one action randomly among the survivors according to a particular distribution.

For estimating μ, Krishnamurthy et al. (2018) used a centering trick on the context vectors $b_i(t)$'s to cancel out $\nu(t)$. They proposed the following estimator for μ:

$$\hat{\mu}(t) = \big(\gamma I_d + \sum_{\tau=1}^{t-1} X_\tau X_\tau^T\big)^{-1}\sum_{\tau=1}^{t-1} X_\tau r_{a(\tau)}(\tau), \tag{8}$$

where $X_\tau = b_{a(\tau)}(\tau) - \mathbb{E}(b_{a(\tau)}(\tau)|\mathcal{F}_{\tau-1})$ and $\gamma > 0$. Given $\mathcal{F}_{\tau-1}$, we see that $\mathbb{E}(X_\tau|\mathcal{F}_{\tau-1}) = 0_d$. Hence, $\{\sum_{\tau=1}^t X_\tau\}_{t=1}^\infty$ is a vector martingale process adapted to filtration $\{\mathcal{F}_t\}_{t=1}^\infty$. Krishnamurthy et al. (2018) derived a $(1-\delta)$-probability upper bound for $b^T(\hat{\mu}(t) - \mu)$ using a new concentration inequality for self-normalized vector-valued martingales established by de la Peña et al. (2009) and de la Peña et al. (2004).

The BOSE algorithm does not require any constraint on the action choice probabilities but achieves a $O(d\sqrt{T}\log(T/\delta))$ regret bound. This bound matches the best known regret bound (3) for linear reward models. However, the action elimination step requires $O(N^2)$ computations at

each round. Also, the distribution used to select the action should satisfy a specific condition to guarantee the $O(d\sqrt{T}\log(T/\delta))$ regret bound. The authors only show that there exists a distribution to satisfy this condition when $N > 2$. The construction of such distribution is not a trivial matter since it requires to solve a convex program with N quadratic conditions at every iteration. Furthermore, the bound of $b^T(\hat{\mu}(t) - \mu)$ is valid under $\gamma \geq 4d\log(9T) + 8\log(4T/\delta)$ when $N > 2$, which can dominate the denominator term of $\hat{\mu}(t)$ when t is small. For example, when $d = 35$ and $T = 1900000$ as in the news article recommendation example in Section 6, $\gamma \geq 2476.8$ if we take $\delta = 0.1$. When γ is set to be a tuning parameter, the BOSE algorithm requires in total two tuning parameters, including ω used in the action elimination step.

4. Proposed Method

In this paper, we propose a new algorithm for the semiparametric reward model (5) which improves on the results of Greenewald et al. (2017) while keeping the framework of the TS algorithm. Our method requires only $O(N)$ computations at each round, while Krishnamurthy et al. (2018) requires $O(N^2)$. An action selection distribution for every N is given and does not need to be solved as in Krishnamurthy et al. (2018). The proposed algorithm uses a new estimator $\hat{\mu}(t)$ for μ which enjoys a tighter high-probability upper bound than (8) without having to deal with any potentially big constant, γ. We prove that the high-probability upper bound of the regret $R(T)$ incurred by the proposed algorithm has the same order as the TS algorithm for linear reward models without restricting the action choice probabilities as in Greenewald et al. (2017).

4.1. Proposed Algorithm

Algorithm 1 Proposed TS algorithm

Set $B = I_d$, $y = 0_d$, $v = (2R+6)\sqrt{6d\log(T/\delta)}$, $\delta \in (0,1)$.
for $t = 1, 2, \cdots, T$ **do**
  Compute $\hat{\mu}(t) = B^{-1}y$.
  Sample $\tilde{\mu}(t)$ from distribution $\mathcal{N}(\hat{\mu}(t), v^2B^{-1})$.
  Pull arm $a(t) = \underset{1\leq i\leq N}{\operatorname{argmax}}\, b_i(t)^T\tilde{\mu}(t)$ and get reward $r_{a(t)}(t)$.
  for $i = 1, \cdots, N$ **do**
    Compute $\pi_i(t) = \mathbb{P}(a(t) = i|\mathcal{F}_{t-1})$.
  end for
  Update B and y:
  $B \leftarrow B + \big(b_{a(t)}(t) - \bar{b}(t)\big)\big(b_{a(t)}(t) - \bar{b}(t)\big)^T$,
  $B \leftarrow B + \sum_{i=1}^N \pi_i(t)\big(b_i(t) - \bar{b}(t)\big)\big(b_i(t) - \bar{b}(t)\big)^T$,
  $y \leftarrow y + 2\big(b_{a(t)}(t) - \bar{b}(t)\big)r_{a(t)}(t)$.
end for

Besides (5), we make the same assumptions as in Section 3, (6) and (7). The proposed Algorithm 1 follows the framework of the TS algorithm with two major modifications: the mean and variance of $\tilde{\mu}(t)$. First, we propose a new estimator $\hat{\mu}(t)$ of μ for the mean of $\tilde{\mu}(t)$:

$$\hat{\mu}(t) = \Big(I_d + \hat{\Sigma}_t + \Sigma_t\Big)^{-1}\sum_{\tau=1}^{t-1} 2X_\tau r_{a(\tau)}(\tau), \qquad (9)$$

where $\hat{\Sigma}_t = \sum_{\tau=1}^{t-1} X_\tau X_\tau^T$ and $\Sigma_t = \sum_{\tau=1}^{t-1}\mathbb{E}(X_\tau X_\tau^T|\mathcal{F}_{\tau-1})$. Compared to (8), we note that the proposed estimator stabilizes the denominator using a new term Σ_t instead of γI_d. As a result, we do not need an extra tuning parameter. Hereinafter, let $\bar{b}(\tau)$ denote $\mathbb{E}(b_{a(\tau)}(\tau)|\mathcal{F}_{\tau-1})$ for simplicity. This term can be calculated as $\bar{b}(\tau) = \mathbb{E}\big(\sum_{i=1}^N I(a(\tau) = i)b_i(\tau)\big|\mathcal{F}_{\tau-1}\big) = \sum_{i=1}^N \pi_i(\tau)b_i(\tau)$, where $\pi_i(\tau) = \mathbb{P}(a(\tau) = i|\mathcal{F}_{\tau-1})$ is the probability of pulling the i-th arm at time τ, which is determined by the distribution of $\tilde{\mu}(\tau)$. Also, the covariance $\mathbb{E}(X_\tau X_\tau^T|\mathcal{F}_{\tau-1})$ can be computed as $\mathbb{E}(X_\tau X_\tau^T|\mathcal{F}_{\tau-1}) = \sum_{i=1}^N \pi_i(\tau)(b_i(\tau) - \bar{b}(\tau))(b_i(\tau) - \bar{b}(\tau))^T$. As for the variance of $\tilde{\mu}(t)$, we propose $v^2B(t)^{-1}$, where $v = (2R+6)\sqrt{6d\log(T/\delta)}$ and $B(t) = I_d + \hat{\Sigma}_t + \Sigma_t$.

In the following theorem, we establish a high-probability regret upper bound for the proposed algorithm.

Theorem 4.1. *Under (5), (6), and (7), the regret of Algorithm 1 is bounded as follows. For $\forall\delta \in (0,1)$, with probability $1-\delta$,*

$$R(T) \leq O\big(d^{3/2}\sqrt{T}\sqrt{\log(Td)\log(T/\delta)}\big(\sqrt{\log(1+T/d)} + \sqrt{\log(1/\delta)}\big)\big).$$

This bound matches the bound (4) of the original TS algorithm for linear reward models. Table 1 compares the properties of the proposed TS algorithm with action-centered TS and BOSE algorithms. The proof of Theorem 4.1 essentially follows the lines of the proof given by Agrawal and Goyal (2013) with some modifications. A complete proof is presented in the Supplementary Material. The main contribution of this paper is a new theorem for the first stage, which bounds $|(b_i(t) - \bar{b}(t))^T(\hat{\mu}(t) - \mu)|$ with high probability with respect to the new estimator (9).

Theorem 4.2. *Let the event $E^{\hat{\mu}}(t)$ be defined as follows:*

$$E^{\hat{\mu}}(t) = \big\{\forall i : \big|\big(b_i(t) - \bar{b}(t)\big)^T(\hat{\mu}(t) - \mu)\big| \leq l(t)s_{t,i}^c\big\},$$

where $s_{t,i}^c = \sqrt{\big(b_i(t) - \bar{b}(t)\big)^T B(t)^{-1}\big(b_i(t) - \bar{b}(t)\big)}$ and $l(t) = (2R+6)\sqrt{d\log(6t^3/\delta)} + 1$. Then for all $t \geq 1$, for any $0 < \delta < 1$, $\mathbb{P}(E^{\hat{\mu}}(t)) \geq 1 - \frac{\delta}{t^2}$.

Table 1. Comparison of the three semiparametric contextual MAB algorithms.

Properties	Action-Centered TS	BOSE	Proposed TS
Restriction on optimal policy	$\pi^*_{\tilde{a}^*(t)}(t) \in [p_{min}, p_{max}]$	None	None
Derivation of $\pi(t)$	from $\tilde{\mu}(t)$	not specified when $N > 2$	from $\tilde{\mu}(t)$
Number of computations	$O(T)$	$O(N^2T)$	$O(NT)$
Tuning parameters	v	ω and γ (when $N > 2$)	v
$R(T)$	$O(Md^{\frac{3}{2}}\sqrt{T}\sqrt{\log(T/\delta)}^3)$	$O(d\sqrt{T}\log(T/\delta))$	$O(d^{\frac{3}{2}}\sqrt{T}\sqrt{\log(T/\delta)}^3)$

4.2. A Sketch of Proof for Theorem 4.2

By decomposition of $(\hat{\mu}(t) - \mu)$,

$$\begin{aligned}\hat{\mu}(t) - \mu &= B(t)^{-1}\sum_{\tau=1}^{t-1} 2X_\tau r_{a(\tau)}(\tau) - \mu \\ &= B(t)^{-1}\Big\{\sum_{\tau=1}^{t-1} 2X_\tau \eta_{a(\tau)}(\tau) \\ &\quad + \sum_{\tau=1}^{t-1} 2X_\tau\big(\nu(\tau) + \bar{b}(\tau)^T\mu\big) - \mu + \sum_{\tau=1}^{t-1} D(\tau)\mu\Big\},\end{aligned}$$

where $X_\tau = b_{a(\tau)}(\tau) - \bar{b}(\tau)$ and $D(\tau) = X_\tau X_\tau^T - \mathbb{E}\big(X_\tau X_\tau^T | \mathcal{F}_{\tau-1}\big)$. Let $b_i^c(t) := b_i(t) - \bar{b}(t)$. Hereinafter, we define $||x||_A := \sqrt{x^T A x}$ for any d-dimensional vector x and any $d \times d$ matrix A. By Cauchy-Schwarz inequality,

$$\big|b_i^c(t)^T(\hat{\mu}(t) - \mu)\big| \le s_{t,i}^c\{2C_1 + 2C_2 + C_3 + C_4\}, \tag{10}$$

where $C_1 = \big|\big|\sum_{\tau=1}^{t-1} X_\tau \eta_{a(\tau)}(\tau)\big|\big|_{B(t)^{-1}}$,

$C_2 = \big|\big|\sum_{\tau=1}^{t-1} X_\tau\big(\nu(\tau) + \bar{b}(\tau)^T\mu\big)\big|\big|_{B(t)^{-1}}$,

$C_3 = \big|\big|\sum_{\tau=1}^{t-1} D(\tau)\mu\big|\big|_{B(t)^{-1}}$,

and $C_4 = \big|\big|\mu\big|\big|_{B(t)^{-1}}$. First, we have $C_4 \le 1$. Now we need to bound C_1, C_2, and C_3. The term C_1 is a familiar term, which we can bound using the technique of Abbasi-Yadkori et al. (2011). Since $\eta_{a(\tau)}(\tau)$ is R-sub-gaussian given $\mathcal{F}_{\tau-1}$ and $a(\tau)$ while X_τ is fixed given $\mathcal{F}_{\tau-1}$ and $a(\tau)$, we have for any $\lambda \in \mathbb{R}^d$,

$$\mathbb{E}\Big[\exp\Big\{\frac{\eta_{a(\tau)}(\tau)}{R}\lambda^T X_\tau - \frac{1}{2}\lambda^T X_\tau X_\tau^T\lambda\Big\}\Big|\mathcal{F}_{\tau-1}, a(\tau)\Big] \le 1.$$

Then it follows,

$$\mathbb{E}\Big[\exp\Big\{\lambda^T\sum_{\tau=1}^{t-1}\frac{\eta_{a(\tau)}(\tau)}{R}X_\tau - \frac{1}{2}\lambda^T\hat{\Sigma}_t\lambda\Big\}\Big] \le 1. \tag{11}$$

From (11), we can apply the following lemma, which is a simplified version of the Corollary 4.3 of de la Peña et al. (2004).

Lemma 4.3. *Let $X_\tau \in \mathbb{R}^d$ and $c_\tau \in \mathbb{R}$ be some random variables, $\tau = 1, \cdots, t$. Suppose $\exists d \times d$ symmetric, positive semi-definite matrix $A(t)$ such that for any $\lambda \in \mathbb{R}^d$,*

$$\mathbb{E}\Big[\exp\Big\{\lambda^T\sum_{\tau=1}^{t} X_\tau c_\tau - \frac{1}{2}\lambda^T A(t)\lambda\Big\}\Big] \le 1. \tag{12}$$

Then for any $0 < \delta < 1$ and any symmetric, positive definite matrix Q, with probability at least $1 - \delta$,

$$\Big|\Big|\sum_{\tau=1}^{t} X_\tau c_\tau\Big|\Big|^2_{(Q+A(t))^{-1}} \le \log\Big(\frac{det\big(Q + A(t)\big)/det\big(Q\big)}{\delta^2}\Big).$$

Taking $c_\tau = \frac{1}{R}\eta_{a(\tau)}(\tau)$, $Q = I_d + \Sigma_t$, and $A(t) = \hat{\Sigma}_t$, we see that (11) corresponds to the condition (12) of the lemma. Also, $C_1 = R\Big|\Big|\sum_{\tau=1}^{t-1} X_\tau c_\tau\Big|\Big|_{(Q+A(t))^{-1}}$. Therefore by Lemma 4.3, for any $0 < \delta < 1$, with probability at least $1 - \frac{\delta}{3t^2}$,

$$C_1 \le R\sqrt{\log\Big(\frac{det(Q + A(t))}{(\delta/(3t^2))^2}\Big)} = R\sqrt{\log\Big(\frac{det(B(t))}{(\delta/(3t^2))^2}\Big)}. \tag{13}$$

Now, we need to bound C_2 and C_3, which are terms that arise due to the $\nu(\tau)$'s and the use of a new estimator (9). Although C_2 looks similar to C_1, the term $(\nu(\tau) + \bar{b}(\tau)^T\mu)$ is not sub-Gaussian, so we can no longer use the technique of Abbasi-Yadkori et al. (2011). Instead, we have $\mathbb{E}[X_\tau|\mathcal{F}_{\tau-1}] = 0$. To bound a similar term to C_2, Krishnamurthy et al. (2018) proposed to use Lemma 7 of de la Peña et al. (2009) for vector-valued martingales to derive an inequality analogous to (11). Using Lemma 7 of de la Peña et al. (2009), we can prove that for any $\lambda \in \mathbb{R}^d$,

$$\mathbb{E}\Big[\exp\Big\{\lambda^T\sum_{\tau=1}^{t-1} X_\tau c_\tau - \frac{1}{2}\lambda^T\big(\hat{\Sigma}_t + \Sigma_t\big)\lambda\Big\}\Big] \le 1. \tag{14}$$

where $c_\tau = \Big(\frac{\nu(\tau) + \bar{b}(\tau)^T\mu}{2}\Big)$. Taking $A(t) = \hat{\Sigma}_t + \Sigma_t$ and $Q = I_d$, (14) corresponds to condition (12). Also, $C_2 = 2\Big|\Big|\sum_{\tau=1}^{t-1} X_\tau c_\tau\Big|\Big|_{(Q+A(t))^{-1}}$. Hence by Lemma 4.3, for any $0 < \delta < 1$, with probability at least $1 - \frac{\delta}{3t^2}$,

$$C_2 \le 2\sqrt{\log\big(det(B(t))/(\delta/3t^2)^2\big)}. \tag{15}$$

The final step is to bound C_3. However, C_3 does not take the form $||\sum X_\tau c_\tau||_{B(t)^{-1}}$, so we require additional work. Let $Y_\tau = D(\tau)\mu$. Then, note that $Y_\tau \in \mathbb{R}^d$ and $\mathbb{E}\big[Y_\tau|\mathcal{F}_{\tau-1}\big] = 0$. We propose the following lemma.

Lemma 4.4. *For any* $\lambda \in \mathbb{R}^d$,

$$\mathbb{E}\Big[\exp\Big\{\lambda^T \sum_{\tau=1}^{t-1} \frac{1}{\sqrt{2}} Y_\tau - \frac{1}{2}\lambda^T\big(\hat{\Sigma}_t + \Sigma_t\big)\lambda\Big\}\Big] \leq 1.$$

The proof of Lemma 4.4 is presented in the Supplementary Material. Taking $A(t) = \hat{\Sigma}_t + \Sigma_t$ and $Q = I_d$, Lemma 4.4 corresponds to condition (12). Also, $C_3 = 2\Big|\Big| \sum_{\tau=1}^{t-1} \frac{1}{\sqrt{2}} Y_\tau \Big|\Big|_{(Q+A(t))^{-1}}$. By Lemma 4.3, for any $0 < \delta < 1$, with probability at least $1 - \frac{\delta}{3t^2}$,

$$C_3 \leq 2\sqrt{\log\big(det(B(t))/(\delta/3t^2)^2\big)}. \tag{16}$$

Plugging the bounds (13), (15) and (16) into (10) completes the proof. For any $0 < \delta < 1$, for all $i = 1, \cdots, N$, with probability at least $1 - \frac{\delta}{t^2}$,

$$\begin{aligned}\big|b_i^c(t)^T(\hat{\mu}(t) - \mu)\big| &\leq s_{t,i}^c\Big\{(2R+6)\sqrt{\log\Big(\frac{det(B(t))}{(\delta/(3t^2))^2}\Big)} + 1\Big\} \\ &\leq l(t)s_{t,i}^c,\end{aligned}$$

where the second inequality is due to the determinant-trace inequality, $det(B(t)) \leq \big(trace(B(t))/d\big)^d \leq (2t)^d$.

5. Simulation Study

We conduct simulation studies to evaluate the proposed algorithm, the original TS algorithm (Agrawal and Goyal, 2013), the action-centered TS (ACTS) algorithm (Greenewald et al., 2017) and the BOSE algorithm (Krishnamurthy et al., 2018). We set N=2 or 6 and d=10. We let the first action to be the base action, i.e., $b_1(t) = 0_d$ for all t, and form the other context vectors as $b_i(t) = [I(i=2)z_i(t)^T, \cdots, I(i=N)z_i(t)^T]^T$, where $z_i(t) \in \mathbb{R}^{d'}$, d'=$d/(N-1)$, and $z_i(t)$ is generated uniformly at random from the d'-dimensional unit sphere. We generate $\eta_i(t) \overset{i.i.d.}{\sim} \mathcal{N}(0, 0.1^2)$ and the rewards from (5), where we set $\mu = [-0.55, 0.666, -0.09, -0.232, 0.244, 0.55, -0.666, 0.09, 0.232, -0.244]^T$ and consider four cases for $\nu(t)$: (i) $\nu(t) = 0$, (ii) $\nu(t) = -b_{a^*(t)}(t)^T\mu$, (iii) $\nu(t) = \log(t+1)$, (iv) $\nu(t) = \cos(t\pi/5000)\log(t+1)$. We conduct 50 replications in total for each case. Note that all four algorithms have one tuning parameter each that controls the degree of exploration. For the TS algorithms, the tuning parameter is v in the variance of $\tilde{\mu}(t)$, and for the BOSE algorithm, ω in the action elimination step. For each algorithm, we use the value of the parameter which incurs minimum median regret. These values can be found by grid search.

Table 2. Median of $R(T)$ over 50 simulations.

Algorithms	N	(i)	(ii)	(iii)	(iv)
TS	2	2.4	57.4	981.2	1724.1
ACTS		20.6	21.3	605.7	1407.5
Proposed TS		19.9	17.4	644.6	1619.3
BOSE		22.7	20.6	657.2	1660.9
TS	6	6.9	74.5	9411.6	11299.2
ACTS		200.1	257.4	3837.8	4592.3
Proposed TS		59.0	30.0	1118.7	2245.3
BOSE		—	—	—	—

Figures 1 and 2 show the cumulative regret $R(t)$ according to time t. The solid lines represent the median values and the dashed lines represent the lower and upper 25% percentiles. The values of $R(T)$ for each algorithm in each case are reported in Table 2. Figure 1 summarizes the results when $N = 2$. When $\nu(t) = 0$, the original TS algorithm achieves lowest cumulative regret. The proposed method shows the second best performance in this case, while the BOSE and ACTS algorithms are also competitive. In cases where $\nu(t)$ changes with time, the original TS algorithm hardly learns at all, while the three other methods developed under the nonparametric intercept term are competitive. When $N = 6$, Figure 2 exhibits a similar trend for the original TS and the proposed algorithms as in the $N = 2$ case. On the other hand, the BOSE algorithm has no explicit method so the results are not shown and the ACTS algorithm has much slower learning speed than the proposed TS method.

6. Real Data Analysis

We present the results of the proposed and existing methods using the R6A dataset provied by Yahoo! Webscope. The dataset is observational log data of user clicks from May 1st, 2009 to May 10th, 2009, which corresponds to 45,811,883 user visits. At every visit, one article was chosen uniformly at random from 20 articles ($N = 20$) and was displayed on the Featured tab of the Today module on Yahoo! front page. The reward $r_i(t)$ is binary, taking value 1 if the visiting user clicked the i-th article, and $r_i(t) = 0$ otherwise. For each article i, there is a context vector $b_i(t) \in \mathbb{R}^{35}$, which is constituted of 5 extracted user features, 5 extracted article features and their products. The extracted features were constructed from high-dimensional raw data for user and article features using a dimension reduction method of Chu et al. (2009).

Evaluating a new reinforcement learning policy retrospectively using observational log data is a challenging task itself and calls for off-policy evaluation methods. This is because in the log data, the rewards of the actions that were not chosen by the original logging policy are missing. In our

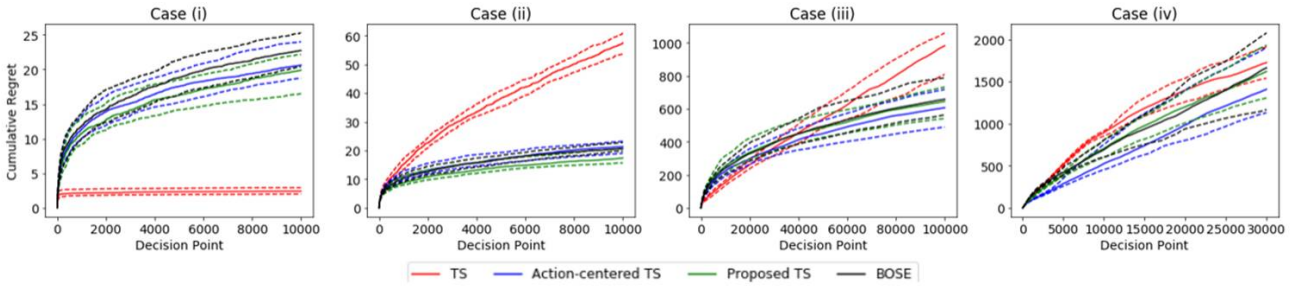

Figure 1. Median (solid), 1st and 3rd quartiles (dashed) of cumulative regret over 50 simulations when $N = 2$.

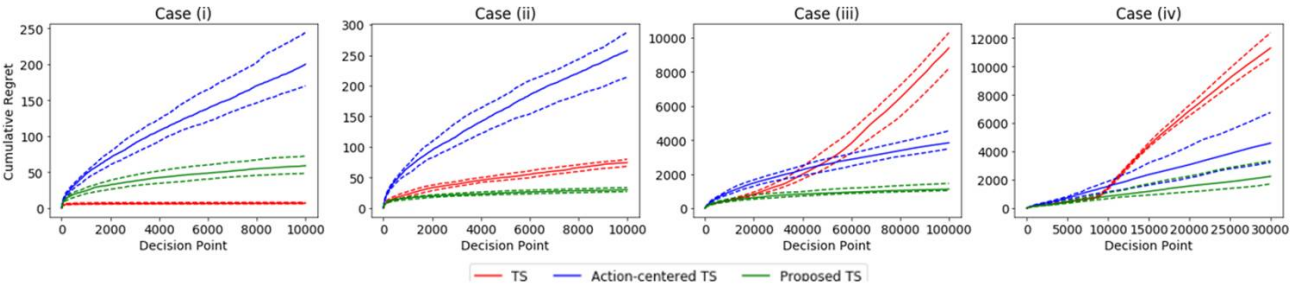

Figure 2. Median (solid), 1st and 3rd quartiles (dashed) of cumulative regret over 50 simulations when $N = 6$.

log data, only the rewards of articles chosen by the uniform random policy are observed and recorded.

Denote the policy that we want to evaluate as $\mathtt{A}$ and the total T-trial reward of $\mathtt{A}$ as $G_{\mathtt{A}}(T) := \mathbb{E}\big[\sum_{t=1}^{T} r_{a(t)}(t)\big]$, where $a(t)$ is chosen by $\mathtt{A}$. Li et al. (2011) proposed an offline evaluation algorithm for estimating $G_{\mathtt{A}}(T)$. Given the stream of events $(\boldsymbol{b} = \{b_i\}_{i=1}^{N}, a, r_a)$ from log data, the algorithm picks up the events of which the chosen action a matches the choice of $\mathtt{A}$ and stacks them into the history of $\mathtt{A}$. Rewards in the history are used to construct the estimate $\hat{G}_{\mathtt{A}}(T)$. In the case where $\mathtt{A}$ is an online learning policy, the history is used to update the action selection distribution of $\mathtt{A}$ as well. Under the condition that $(\boldsymbol{b}(t), r(t))$ are i.i.d. and the logging policy is the uniform random policy, $\hat{G}_{\mathtt{A}}(T)$ is shown to be unbiased. We note that the i.i.d. condition does not cover the case where $\nu(t)$ is adaptive to the past trials.

We evaluate the uniform random policy, TS algorithm and the proposed algorithm using the method of Li et al. (2011). We use data of May 1st, 2009 as tuning data to choose the optimal exploration parameter v for the TS algorithm and the proposed algorithm, respectively. Then we conduct main analysis on data from May 2nd to May 10th, 2009. Note that the method of Li et al. (2011) picks up over $1/N = 1/20$ of the log data. This corresponds to over $T = 1900000$. We fix the value of T to $T = 1900000$ a priori, and conduct the evaluation algorithm for 10 times on the same data for each policy. Since the evaluated policies are all randomized algorithms, each of the 10 runs pick up different actions, giving 10 different estimates. We report the mean, 1st quartile and 3rd quartile of the estimates for each policy in Table 3. We verify that the contextual bandit algorithms achieve substantially higher user click rates than the uniform random policy. Among the contextual bandit algorithms, the proposed algorithm increases the average user click rate by 4.4% compared to the original TS algorithm.

Table 3. Mean, 1st quartile (1st Q.) and 3rd quartile (3rd Q.) of user clicks achieved by each policy over 10 runs.

Policies	Mean	1st Q.	3rd Q.
Uniform policy	66696.7	66515.0	66832.8
TS algorithm	86907.0	85992.8	88551.3
Proposed TS	90689.7	90177.3	91166.3

7. Concluding Remarks

This paper proposes a new contextual MAB algorithm for a semiparametric reward model which is well suited to real problems where baseline rewards are bound to change with time. The proposed algorithm improves on existing methods that consider the same model. Simulation study and real data analysis demonstrate the advantage of the proposed method.

Acknowledgements

The authors were supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIP) (No.2017R1A2B4008956).

References

Abbasi-Yadkori, Y., Pál, D. and Szepesvári, C. Improved algorithms for linear stochastic bandits. In *Advances in Neural Information Processing Systems,* pp. 2312–2320, 2011.

Agrawal, S. and Goyal, N. Thompson sampling for contextual bandits with linear payoffs. In *Proceedings of the 30th International Conference on Machine Learning*, pp. 127–135, 2013.

Auer, P. Using confidence bounds for exploitation-exploration trade-offs. *Journal of Machine Learning Research,* 3:397–422, 2002.

Beygelzimer, A., Langford, J., Li, L., Reyzin, L., and Schapire, R.E. Contextual bandit algorithms with supervised learning guarantees. In *Proceedings of the 14th International Conference on Artificial Intelligence and Statistics*, pp. 19–26, 2011.

Chu, W., Park, S.T., Beaupre, T., Motgi, N., Phadke, A., Chakraborty, S. and Zachariah, J. A case study of behavior-driven conjoint analysis on Yahoo!: front page today module. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining,* pp. 1097-1104, 2009.

Chu, W., Li, L., Reyzin, L. and Schapire, R.E. Contextual bandits with linear payoff functions. In *Proceedings of the 14th International Conference on Artificial Intelligence and Statistics,* pp. 208–214, 2011.

Dani, V., Hayes, T. P. and Kakade, S. M. Stochastic linear optimization under bandit feedback. In *Conference on Learning Theory*, pp. 355-366, 2008.

de la Peña, V. H., Klass, M. J. and Lai, T. L. Self-normalized processes: exponential inequalities, moment bounds and iterated logarithm laws. *Annals of Probability,* 32(3A):1902–1933, 2004.

de la Peña ,V. H., Klass, M. J. and Lai, T. L. Theory and applications of multivariate self-normalized processes. *Stochastic Processes and their Applications,* 119(12):4210–4227, 2009.

Even-Dar, E., Mannor, S. and Mansour, Y. Action elimination and stopping conditions for the multi-armed bandit and reinforcement learning problems. *Journal of machine learning research*, 7:1079-1105, 2006.

Ferreira, K., Simchi-Levi, D. and Wang, H. Online network revenue management using Thompson sampling. *Operations Research,* 66(6):1586–1602, 2018.

Greenewald, K., Tewari, A., Murphy, S. and Klasnja, P. Action centered contextual bandits. In *Advances in Neural Information Processing Systems*, pp. 5977–5985, 2017.

Kawale, J., Bui, H.H., Kveton, B., Tran-Thanh, L. and Chawla, S. Efficient Thompson sampling for online matrix-factorization recommendation. In *Advances in Neural Information Processing Systems*, pp. 1297–1305, 2015.

Krishnamurthy, A., Wu, Z. S. and Syrgkanis, V. Semiparametric contextual bandits. In *Proceedings of the 35th International Conference on Machine Learning,* 2018.

Lai, T.L. and Robbins, H. Asymptotically efficient adaptive allocation rules. *Advances in Applied Mathematics,* 6(1):4–22, 1985.

Langford, J., Strehl, A. and Wortman, J. Exploration scavenging. In *Proceedings of the 25th International Conference on Machine Learning,* pp. 528–535, 2008.

Li, L., Chu, W., Langford, J. and Schapire, R. E. A contextual-bandit approach to personalized news article recommendation. In *Proceedings of the 19th International Conference on World wide web*, pp. 661–670, 2010.

Li, L., Chu, W., Langford, J. and Wang, X. Unbiased offline evaluation of contextual-bandit-based news article recommendation algorithms. In *Proceedings of the 4th ACM International Conference on Web search and data mining,* pp. 297–306, 2011.

Robbins, H. Some aspects of the sequential design of experiments. *Bulletin of the American Mathematical Society*, 58(5):527–535, 1952.

Schwartz, E.M., Bradlow, E.T. and Fader, P.S. Customer acquisition via display advertising using multi-armed bandit experiments. *Marketing Science,* 36(4):500–522, 2017.

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

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

Yahoo! Webscope. Yahoo! Front Page Today Module User Click Log Dataset, version 1.0. http://webscope.sandbox.yahoo.com. Accessed: 09/01/2019.