Comments on "Semi-Parametric Sampling for Stochastic Bandits with Many Arms"

MARTYNA KSYTA

In "Semi-parametric sampling for stochastic bandits with many arms" [35] the authors introduced a new contextual multi-armed bandit algorithm (called LSPS). It was claimed to be superior to the existing methods in situations with large number of candidate arms, when the reward depends on a context and arm-specific factor. The results were supported by better Bayesian regret bound and experimental performance.

In this comment, we raise serious concerns over the formulation of Bayesian regret bound presented in the work mentioned above - it is a decreasing function of number of steps and depends on a random variable associated with the estimated parameters. We also point and fix (with a proof) a minor mistake in a provided lemma. Additionally we reproduce experiments on synthetic data originally performed by the authors. Based on that we conclude that LSPS does not always outperform simpler models.

CCS Concepts: • **Applied computing** → **Online shopping**; • **Computing methodologies** → **Machine learning algorithms**; **Simulation evaluation**.

Additional Key Words and Phrases: recommender system, multi-armed bandit, Thompson sampling

ACM Reference Format:
Martyna Ksyta. 2020. Comments on "Semi-Parametric Sampling for Stochastic Bandits with Many Arms". 1, 1 (October 2020), 14 pages.

1 INTRODUCTION

1.1 Multi-armed bandit

Multi-armed bandit is an optimization problem. There is an agent and a set of possible actions (also called arms) she can choose from at each time step from 1 to T. An action brings a reward, which is revealed only after its selection. Agent's goal is to maximize the sum of rewards for all the time steps. In the contextual version of the problem before deciding on an action the agent also knows the context - feature vectors associated with the arms. Is it assumed that the reward is a function or a parametrized random variable. As time passes the agent knowns more about the reward. At each moment she can decide to what extend use already gathered information (i.e. selecting arms with the context that has previously yielded high rewards) and to what extend explore (i.e. choosing a new context or arms) in the hope of finding arms giving even better rewards. This is a classical reinforcement learning problem known as the exploration–exploitation trade-off. The detailed assumptions about arms, rewards and the context depend on the algorithm. The work commented by us focus on a situation with the context when a number of arms to choose from might be large - in the next subsection we will mention work and assumption behind the models related to this problem.

Author's address: Martyna Ksyta, ksyta.martyna@gmail.com.

Manuscript submitted to ACM

1.2 Related work

There is a range of algorithms that do not take context into account assuming the rewards are independent between arms. A model presented in [5] uses parametrized random variables as rewards of the arms. Some prior distributions on the parameters are assumed. At any time step an arm is selected according to its posterior probability of being the best one. This algorithm is known in general as Thompson sampling [46] and it was empirically proved to be effective by multiple studies ([21], [43], [20], [10], [34], [25]), as stated in [5]. Other solutions not incorporating context comprise of Upper Confidence Bound algorithms studied by [30], [8], [9], [24] and [3].

The disadvantage of such models is that they do not incorporate information about the context, which might be shared between arms. There are algorithms taking this into account. LinUCB [31], analyzed later in [11], is based on the assumption that expected rewards are linear functions of the context, separate for each arm. With a prior distribution of their coefficient vectors we can derive formula for confidence bound of the expected reward for each arm and select the one maximizing it. [4] assume also a linear model but with a single coefficient vector shared between arms and distributed normally. The difference between arms is incorporated using feature vectors. The selected arm maximizes the reward calculated with the coefficient vector sampled from its posterior distribution. Other models based on the assumption that regret depends linearly on the context, were presented in [7], [13], [39], [1] and [26], which deals with the high-dimensional context. There are also contextual multi-armed bandit algorithms based on general linear models studied by [16], [32], [23] and [29].

Assumption about linearity of the regret might not always be feasible. Recently non-linear bandit algorithms were proposed. There are models based on neural networks: [6], [12], random forests: [15], decision trees [14], piecewise constant estimators of the reward function: [38], [44], [37] and Gaussian processes: [27], [45]. There is also a kernelized non-linear version of UCB: [47]. The algorithm presented in [17] enables use of many predictor classes e.g. regularized linear functions or gradient-boosted regression trees. There are also attempts to combine multi-armed bandit with clustering [18], [33] and [48]. Authors of [22] consider situation when the rewards are known functions of a common latent random variable. There is an algorithm testing linearity of regret and then selecting appropriate model proposed by [19].

To overcome the discrepancy between reality and model assumptions the semi-parametric models were also introduced by [28] and [35]. These models assume that the reward for an arm come from a distribution whose mean is determined by two factors: one dependent on the features of an arm (parametric part) and the other dependent only on an arm (not-parametric part). In [36] the authors present a semi-parametric model in which the features depend also on the time step.

2 LINEAR SEMI-PARAMETRIC SAMPLING MODEL

In order to establish the notation for further use, we present below the model described in [35], called LSPS.

2.1 Assumptions

The agent has to select an action for each time step $t \in 1, 2, \ldots, T$. The possible actions are fixed over time and indexed by $i = 1, 2, \ldots, N$. There is a feature vector $x_i \in \mathbb{R}^d$, $||x_i|| \leq 1$, associated with each action. The reward for an action, denoted as r_i, is a random variable. Each time an agent selects i-th action the observed reward is sampled from r_i. r_i may be represented as its expected value - γ_i - disturbed by a random noise η_t. Values of γ_i are initially and only once sampled for each arm and do not change over time. The means of their distributions are linear functions of x_i and coefficient vector θ, which must be found by the algorithm. Conditionally on θ variables γ_i are independent. There is a prior distribution of θ assumed. Values σ_1, σ_2, $\sigma_3 \in \mathbb{R}$ are hyperparameters of the model. All these assumptions, together with the distributions used by the authors, are summarized below:

$$r_i|\gamma_i \sim \mathcal{N}(\gamma_i, \sigma_1^2)$$
$$\gamma_i|(\theta, x_i) \sim \mathcal{N}(\theta^T x_i, \sigma_2^2)$$
$$\theta \sim \mathcal{N}(0, \sigma_3^2 I)$$

2.2 Thompson sampling

In order to select an arm at time t the LSPS algorithm uses Thompson sampling. It is described in the frame below. We follow a similar notation to [42].

Algorithm 1 Thompson sampling in LSPS

$H_t \leftarrow \emptyset$
for $t = 1, 2, \ldots, T$ **do**
  sample $\theta_t \sim \mathbb{P}(\theta|H_t)$
  for $i = 1, 2, \ldots, N$ **do**
    sample $\gamma_{i,t} \sim \mathbb{P}(\gamma_{i,t}|\theta_t, H_t)$
  end for
  $i_t \leftarrow \text{argmax}_{i=1,2,\ldots,N}\ \gamma_{i,t}$
  observe reward $r_{i_t,t}$ for i_t
  $H_t \leftarrow H_t \cup \{(i_t, r_{i_t,t})\}$
end for

2.3 Posterior distributions

In order to use Thompson sampling one must know $\mathbb{P}(\theta|H_t)$ and $\mathbb{P}(\gamma_{i,t}|\theta_t, H_t)$. As calculated in [35] these distributions depends on:

$n_{i,t}$ - number of times the i-th arm was chosen up to time t,
$\bar{r}_{i,t}$ - average observed reward of the i-th arm up to time t

The posterior distribution of θ can be represented as follows:

$$\theta|H_t \sim \mathcal{N}(\widehat{\theta}_t, A_t^{-1}) \tag{1}$$

where:

$$\widehat{\theta}_t = A_t^{-1} b_t$$
$$A_t = \frac{1}{\sigma_3^2} I + \sum_{i=1}^{N} \frac{n_{i,t}}{\sigma_1^2 + n_{i,t}\sigma_2^2} x_i x_i^T$$
$$b_t = \sum_{i=1}^{N} \frac{n_{i,t}\bar{r}_{i,t}}{\sigma_1^2 + n_{i,t}\sigma_2^2} x_i$$

The formula for posterior distribution of γ_i is:

$$\gamma_i|(\theta_t, H_t) \sim \mathcal{N}(\widehat{\gamma}_{i,t}, \sigma_{i,t}^2) \tag{2}$$

where:

$$\widehat{\gamma}_{i,t} = \frac{\sigma_2^2 n_{i,t}\overline{r}_{i,t} + \sigma_1^2\theta_t^T x_i}{\sigma_1^2 + n_{i,t}\sigma_2^2}$$

$$\sigma_{i,t}^2 = \frac{\sigma_1^2\sigma_2^2}{\sigma_1^2 + n_{i,t}\sigma_2^2}$$

3 USEFUL DEFINITIONS

Before presenting our remarks to [35] we would like to start with useful definitions with some explanatory comments.

3.1 Bayesian regret

Bayesian regret is a measure of performance of posterior sampling algorithms. It was first proposed in [40]. It is based on a regret - the difference between the expected reward of the optimal arm and the expected reward of selected arms. For LSPS it can be written in the following way:

$$\text{Regret}(X, \theta, \gamma_1, \ldots\gamma_N, T, N, \sigma_1, \sigma_2, \sigma_3) = \sum_{t=1}^{T} \mathbb{E}[\max_{i=1,\ldots,N} \gamma_i - r_{i_t,t}|\theta, \gamma_1, \ldots\gamma_N]$$

Bayesian regret is defined as the expected value of the regret calculated over the prior distribution of the parameters of the model. For LSPS we get:

$$\text{BayesRegret}(X, T, N, \sigma_1, \sigma_2, \sigma_3) = \mathbb{E}[\text{Regret}]$$

It is a function of X (which can be bound), time T, N, d and hyperparameters of the model: σ_1, σ_2 and σ_3. In [35] there is an additional parameter introduced, which does not fit into this formula.

3.2 R-sub-Gaussian random variable

A random variable η is called R-sub-Gaussian, if $\forall\lambda \geq 0$:

$$\mathbb{E}[e^{\lambda\eta}] \leq e^{\lambda^2 R^2/2}$$

3.3 Notation $\widetilde{O}$

f, g - real valued functions, $g(x) > 0$.
If $\exists k > 0$ so that $f(x) = O\left(g(x) log^k(g(x))\right)$, we can write $f(x) = \widetilde{O}\,(g(x))$.

4 MAIN REMARKS

4.1 Comments on the Bayesian regret bound

In the Theorem 1 the authors of [35] provide the formulation of the Bayesian regret bound of the LSPS algorithm. We will first present the theorem and then raise our concerns about it.

Theorem 1. *If $\forall t \leq T$, η_t is R-sub-Gaussian, $d \leq \sqrt{N}$ and $\epsilon_{max} \leq \sqrt{\frac{N}{dT}\left(\frac{d}{\sqrt{N}}\right)^{2\alpha}}$, where $\alpha \in [0, 1]$, then the algorithm LSPS can achieve upper Bayesian regret bound:*

$$\widetilde{O}(\sqrt{N}^{1-\alpha} d^{\alpha} \sqrt{T}) \tag{3}$$

with

$$\frac{\sigma_2^2}{\sigma_1^2} = \frac{T}{N}\left(\frac{d}{\sqrt{N}}\right)^{\alpha} \tag{4}$$

4.1.1 *Formula for the bound of the Bayesian regret.*

We can use (4) to substitute $\left(\frac{d}{\sqrt{N}}\right)^{\alpha}$ into the formula for the Bayesian regret bound, getting rid of α. This would give us the formula below, which is a decreasing function of T.

$$\widetilde{O}\left(\frac{\sigma_2^2}{\sigma_1^2} N^{\frac{3}{2}} T^{-\frac{1}{2}}\right) \tag{5}$$

This can not be true. The detailed proof is provided in the Appendix. Intuitively the differences between expected rewards of an optimal arm and the selected arms are non-negative for each time step, so their sum can not decrease in time to zero, as the bounding function.

4.1.2 *Impact of ϵ_{max}.*

We would also like to comment the presence of ϵ_{max}. Firstly it is a random variable and its presence in the formulation of the Bayesian regret bound seems inappropriate. According to the definition presented by [35] in the section "Problem Formulation" in equation (1) we can interpret it in the following way: For i-th arm the variable ϵ_i is the difference between parametric part of the expected reward and the expected reward itself. ϵ_{max} is the maximal absolute difference of these values over all arms:

$$\epsilon_i = f(\theta, x_i) - \gamma_i$$

$$\epsilon_{max} = \max\{|\epsilon_i|, \ldots, |\epsilon_N|\}$$

The Bayesian regret bound should be the expected value calculated over the prior distribution of θ. The detailed derivation of the results could cast some light on the reason for presence of ϵ_{max}. It is supposedly present in the supplementary materials. However these materials were not published anywhere and we were not able to obtain them from the Authors. Presence of ϵ_{max} could be explained by the fact, that LSPS is semi-parametric and the latter expectation is calculated conditionally on θ. However the definition of the Bayesian regret presented by the authors of [35] in equation (5) is similar to the one defined in Section 3.1 and it does not assume some fixed parameters. If the definition of Bayesian regret was different for semi-parametric models it would not be possible to compare them to other Thompson sampling models.

Secondly the constraint on ϵ_{max} depends on T. Using similar substitution as for the regret bound we get:

$$\epsilon_{max} \leq \frac{\sigma_2^4}{\sigma_1^4} d^{-1} N^{\frac{5}{2}} T^{-\frac{5}{2}}$$

so we see that the bound for ϵ_{max} narrows with increasing T. This limits the applicability of the Theorem 1. To consider how significant it is effect we can use the attributes of the e-commerce dataset presented by [35], as they should reflect the real-world scenario. In this case $d = 5$, $N = 1000$ and $T = 200000$. The bound for ϵ_{max} equals $3.5 * 10^{-7} \frac{\sigma_2^4}{\sigma_1^4}$, which seem to be low, but in order to validate its sensibility one must know the order of magnitude of σ_1, σ_2 and X.

4.1.3 Claims about α coefficient. We can notice that the claim that $\alpha \in [0,1]$, is not always true. From the equation (4) we see that:

$$\alpha = \frac{\ln T - \ln N + 2(\ln \sigma_1 - \ln \sigma_2)}{\frac{1}{2}\ln(N) - \ln d}$$

From the fact, that the denominator is greater than 0 (because $d \leq \sqrt{N}$), we get that α can be negative if $\frac{T}{N} \leq \frac{\sigma_2^2}{\sigma_1^2}$. This may happen in a situation when N is comparable to T. This situation is described by the authors as one where "LSPS can achieve significant improvement even when bias is relatively large". In such case α is smaller than 0 for σ_2 accordingly greater than σ_1. If $\alpha < 0$, the $(\sqrt{N}/d)^{\alpha}$ improvement declared by the authors is actually a deterioration. To prevent this we need additional assumption that $\frac{T}{N} \geq \frac{\sigma_2^2}{\sigma_1^2}$.

Value of α can be also > 1 if:

$$T > \frac{N\sqrt{N}}{d}\left(\frac{\sigma_2}{\sigma_1}\right)^2$$

This however does not have negative impact on the Theorem 1.

4.2 Comments on a lemma

The following claim is stated in [35] as Lemma 1:

Lemma 1. *For any $t \leq T$, with probability* $1 - \frac{\delta}{NT^2}$

$$|\gamma_{i,t} - \widehat{\gamma_{i,t}}| \leq \sqrt{2\ln\frac{NT^2}{2\delta}}\sigma_{i,t} \tag{6}$$

This lemma is not completely correct. Taking $\frac{\delta}{NT^2} = \frac{1}{2}$ we get, that with probability $\frac{1}{2}$:

$$|\gamma_{i,t} - \widehat{\gamma_{i,t}}| \leq 0$$

which is equivalent to:

$$\mathbb{P}\left(\gamma_{i,t} = \widehat{\gamma_{i,t}}\right) = \frac{1}{2}$$

From the model formulation in (2) we know that:

$$\gamma_i|(\theta_t, H_t) \sim \mathcal{N}(\widehat{\gamma}_{i,t}, \sigma_{i,t}^2)$$

so we can calculate:

$$\mathbb{P}\left(\gamma_{i,t} = \widehat{\gamma_{i,t}}\right) = \mathbb{E}[\mathbb{P}(\gamma_{i,t} = \widehat{\gamma_{i,t}}|\,(\theta_t, H_t))] = \mathbb{E}[0|\,(\theta_t, H_t)] = 0 \neq \frac{1}{2}$$

The lemma is true if we use: $\sqrt{2\ln\frac{NT^2}{\delta}}\sigma_{i,t}$ in the right-hand side of the inequality (6) and require the probability to be at least $1 - \frac{\delta}{NT^2}$.

Proof. Let's denote $a = \sqrt{2\ln\frac{NT^2}{\delta}}$ and $X = \frac{\gamma_{i,t} - \widehat{\gamma_{i,t}}}{\sigma_{i,t}}$. We know that:

$$X|(\theta_t, H_t) \sim \mathcal{N}(0,1)$$

so we can apply Proposition 1 to get:

$$\mathbb{P}\left(|X| \leq a|(\theta_t, H_t)\right) \geq 1 - e^{-\frac{a^2}{2}}$$

After taking the expectation of both sides we get the formula to be proven. □

Proposition 1. *For any $a > 0$ and $X \sim \mathcal{N}(0,1)$:*

$$\mathbb{P}(|X| \leq a) \geq 1 - e^{-\frac{a^2}{2}}$$

Proof. The density of normal distribution is an even function and $a > 0$, so we can write:

$$\mathbb{P}(|X| > a) = 2\mathbb{P}(X > a) = 2\int_a^{+\infty} \frac{1}{\sqrt{2\pi}} e^{-x^2/2} dx = \frac{2}{\sqrt{\pi}} \int_{\frac{a}{\sqrt{2}}}^{+\infty} e^{-t^2} dt$$

The integral can be bounded using Propositions 2 and 3:

$$\frac{2}{\sqrt{\pi}} e^{-\frac{a^2}{2}} e^{\frac{a^2}{2}} \int_{\frac{a}{\sqrt{2}}}^{+\infty} e^{-t^2} dt \leq \frac{2}{\sqrt{\pi}} e^{-\frac{a^2}{2}} \frac{1}{\frac{a}{\sqrt{2}} + \sqrt{\frac{a^2}{2} + \frac{4}{\pi}}} < e^{-\frac{a^2}{2}}$$

So we finally got that:

$$\mathbb{P}(|X| > a) \leq e^{-\frac{a^2}{2}}$$

and

$$\mathbb{P}(|X| \leq a) = 1 - \mathbb{P}(|X| > a) \geq 1 - e^{-\frac{a^2}{2}}$$

□

Proposition 2. *Formula 7.1.13 from [2]*
For any $z \geq 0$:

$$e^{z^2} \int_z^{+\infty} e^{-t^2} dt \leq \frac{1}{z + \sqrt{z^2 + \frac{4}{\pi}}}$$

Proposition 3. *For any $a > 0$:*

$$\frac{2}{\sqrt{\pi}\left(\frac{a}{\sqrt{2}} + \sqrt{\frac{a^2}{2} + \frac{4}{\pi}}\right)} < 1$$

Proof. The expression is decreasing with respect to a, so for any $a > 0$ is smaller than for $a = 0$:

$$\frac{2}{\sqrt{\pi}\sqrt{\frac{4}{\pi}}} = 1$$

□

4.3 Not related theorem

In the section 5 "Regret analysis" [35] refer to the Proposition 3 from [41]. However this proposition is not connected to authors' statements. We suspect that authors had in mind Proposition 2 from [40], which gives the Bayesian regret bound of $\widetilde{O}\left(\sqrt{NT}\right)$ if rewards are within interval $[0,1]$. We can also find in [40] claim that this proposition can be extended to cases where reward is not bounded but where instead its distribution is "light-tailed". In this case it might fit the statement of [35].

5 EXPERIMENT

We have reproduced the experiment performed by [35] on the synthetic data. The code can be found in this Github repository, files parametrizing the experiments and their results, with saved states of the models and rewards are stored on Google Drive here. All the models were implemented from scratch, since the code was not provided with the publication.

5.1 Hyperparameters

The LSPS and one of the models used for the comparison utilise hyperparameters - their values used in the original experiment were not provided by the authors. They only mentioned to use the same corresponding hyperparameters of the LSPS and TS-Lin models. Because of this we decided to perform grid search over values from $[0.1, 1, 10]$ for reduced number of timesteps ($T = 25000$) and later run models once again with the best hyperparameters for all the timestemps.

We were not able to perform the experiment on e-commerce dataset, because it is not publicly available. We feel that even contradicting results on sythetic dataset are worth sharing with the machine learning community.

5.2 Compared models

LSPS was compared to the models listed below. All of them are based on Thompson sampling.

- TS-Gau [5]. This model does not use the context. It assumes that the reward of each arm $\sim \mathcal{N}(\mu_i, 1)$. The prior distribution of μ_i is $\mathcal{N}(0, 1)$. The posterior distribution is calculated based on the rewards observed for each arm.
- TS-Beta [5]. This is similar model to TS-Gau. The difference is in the used distributions: reward for each arm is assumed to follow the Bernoulli distribution. Its mean has $\mathcal{B}(1, 1)$ prior distribution.
- TS-Lin [4] This model assumes that the expected reward for each arm $\sim \mathcal{N}(x_i^T \mu, v^2)$. The prior of μ is $\mathcal{N}(0, v^2 I)$ and v is a hyperparameter.

5.3 Setup

We have used the same setup as one described in [35] whenever possible. All x_i and θ were sampled from $\mathcal{N}(0, I)$, then we took the absolute values over all of their elements. Later x_i were divided all by the largest of their norms, so $\forall i$ $||x_i|| \leq 1$. Parameter θ was normalized and then multiplied by a predefined constant a. The expected rewards for each arm – γ_i – were set to $x_i^T \theta + \epsilon_i$. Values of ϵ_i were sampled from $\mathcal{U}[0, 1 - a]$. [35] claim to use $\mathcal{U}[1 - a, 1]$ instead, but it is not feasible. In this case we could get values of expected reward higher than 1. The authors claims to model reward using Gaussian and binomial distributions. For the binomial distribution (which has two parameters: p and n) n was not specified, so we assumed that the authors meant $n = 1$, which is equivalent to Bernoulli distribution and in line with the assumption of TS-Beta model. In this case the expected rewards can not be greater than 1. Also if we use parameter a to control the impact of bias ϵ_i on the expected reward, the norm of θ should increase and the possible values of ϵ_i should decrease with increasing a. This is not the case if we use $\mathcal{U}[1 - a, 1]$.

The experiment was performed for 1000 arms, 5-dimensional feature vectors and a equal 0.5 (reflecting semi-parametric case) and 1 (purely linear case). We repeated calculations for 3 values of random seeds: $1, 2, 3$ for hyperparameters tuning and 3 different values for final calculations: $4, 5, 6$. This was not originally performed by the authors.

5.4 Results

We share results from all four cases tested in the simulation. In Figure 1 there is a cumulative regret shown for Gaussian and binomial distributions. Bold lines represent values averaged over all seeds. The shaded areas reflect maximal and minimal values. The cumulative regret of LSPS in some cases is very close to TS-Lin. In order to compare these models we include numerical results in Table 1. [35] present some results on graphs. It is not mentioned which distribution they refer to, so we can not compare all the results directly. However in all cases presented by the authors LSPS is the best model. For $a = 0.5$ the difference is clear, for $a = 1$ is much smaller. We obtained similar results only in case of Gaussian distribution. For $a = 0.5$ LSPS was clearly the best. For $a = 1$ TS-Lin achieved lower cumulative regret than LSPS but its value was within bounds formed by minimum and maximum of LSPS results. Results for binomial distribution show much more variability – for $a = 0.5$ the best model is TS-Lin and the range of cumulative regret for LSPS is very wide. For $a = 1$ LSPS gives the best performance. In order to get certainty which of these models: LSPS or TS-Lin is better more experiments are needed. It can not be decided based only on one run of the simulation, as done in [35].

Another interesting difference between our and the original results is the performance of TS-Lin in the semi-parametric case ($a = 0.5$). [35] shows that it is significantly worse comparing to the other models. We did not get such results. It could be explained by the fact that TS-Lin is able to quickly estimate the rewards of the best arms even in semi-parametric environment. Then it selects them, obtaining more observations and improving the fit for these best arms, at the expense of fit to other ones.

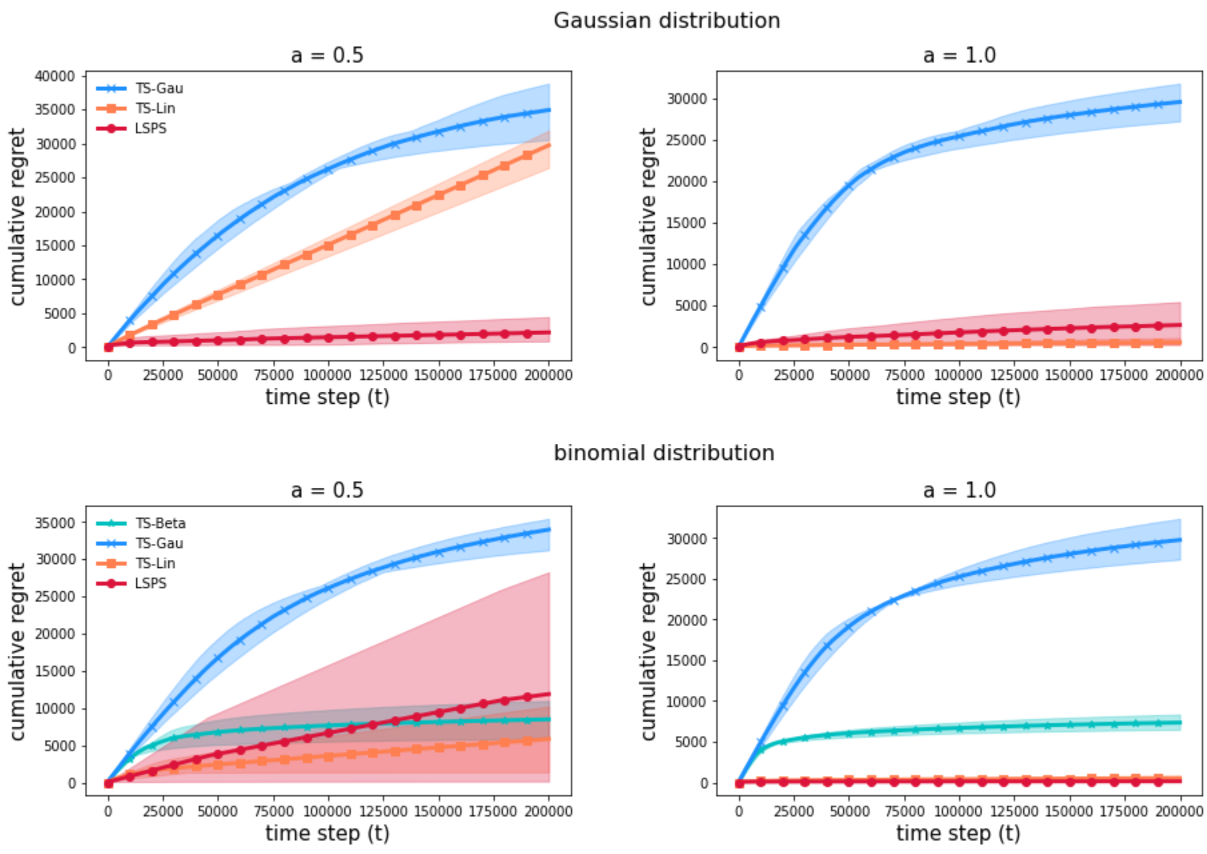

Fig. 1. Results of the experiment

Table 1. Comparison of the models on the synthetic data

Model	Cumulative regret			Hyperparameters
	Mean	Min	Max	
	reward Gaussian, $a = 0.5$			
TS-Gau	34950.22	30551.22	38854.90	-
TS-Lin	29757.08	26442.41	31937.36	v = 1.0
LSPS	**2180.79**	883.14	4445.42	$\sigma_1 = 1.0, \sigma_2 = 0.1, \sigma_3 = 0.1$
	reward Gaussian, $a = 1.0$			
TS-Gau	29532.66	27199.17	31751.64	-
TS-Lin	**587.06**	249.70	1121.74	v = 1.0
LSPS	2675.45	370.63	5452.13	$\sigma_1 = 1.0, \sigma_2 = 0.1, \sigma_3 = 0.1$
	reward binomial, $a = 0.5$			
TS-Beta	8485.57	5918.24	10979.35	-
TS-Gau	33996.48	31228.52	35459.35	-
TS-Lin	**5850.11**	1392.79	10219.05	v = 1.0
LSPS	11887.84	146.80	28273.94	$\sigma_1 = 0.1, \sigma_2 = 0.1, \sigma_3 = 0.1$
	reward binomial, $a = 1.0$			
TS-Beta	7355.23	6493.95	8359.40	-
TS-Gau	29788.95	27389.01	32396.54	-
TS-Lin	537.31	241.48	818.74	v = 1.0
LSPS	**143.27**	80.40	224.21	$\sigma_1 = 0.1, \sigma_2 = 0.1, \sigma_3 = 1.0$

6 APPENDIX

6.1 Bound of the Bayesian regret

Here we will show that the bound for the Bayesian regret provided by [35] cannot be correct. First we will notice that the Bayesian regret is non-decreasing function of time horizon T. Later we will show that the proposed formula for the bound approaches 0 as T goes to infinity. These two facts contradict each other, except for the situation when Bayesian regret is always 0. We will show that it is not the case for LSPS algorithm. In order to simplify the notation we will omit all arguments of BayesRegret except time and write BayesRegret(T).

PROPOSITION 4. *For any $T \geq 1$:*
$BayesRegret(T) \geq 0$ and $BayesRegret(T)$ is non-decreasing function of T.

PROOF.

$$\text{Regret}(T) = \sum_{t=1}^{T} \mathbb{E}[\max_{i=1,\dots,N} \gamma_i - r_{i,t}|\theta, \gamma_1, \dots\gamma_N] = \sum_{t=1}^{T} \mathbb{E}[\max_{i=1,\dots,N} \gamma_i|\theta, \gamma_1, \dots\gamma_N] - \mathbb{E}[r_{i,t}|\theta, \gamma_1, \dots\gamma_N] =$$
$$\sum_{t=1}^{T} \max_{i=1,\dots,N} \gamma_i - \mathbb{E}[r_{i,t}|\theta, \gamma_1, \dots\gamma_N] = \sum_{t=1}^{T} \max_{i=1,\dots,N} \gamma_i - \gamma_{i_t}$$

Each element of the sum is greater or equal 0. Its expected value is also greater or equal 0. The Bayesian regret is sum of such T expected values, so it is non-decreasing function of T and BayesRegret$(T) \geq 0$ for any T. □

PROPOSITION 5. *For any $k > 0$ and any $\alpha > 0$:*

$$\lim_{T\to+\infty} \alpha T^{-\frac{1}{2}} log^k(\alpha T^{-\frac{1}{2}}) = 0$$

PROOF. Let $u = \frac{T^{\frac{1}{2}}}{\alpha}$. Then the limit is equivalent to:

$$\lim_{u\to+\infty} \frac{log^k(u^{-1})}{u} = \lim_{u\to+\infty} \frac{(-1)^k log^k(u)}{u} \overset{\text{H}}{=} \lim_{u\to+\infty} \frac{(-1)^k k log^{k-1}(u)}{u}$$

If $k - 1 \leq 0$ we can rewrite it as:

$$\lim_{u\to+\infty} \frac{(-1)^k k}{u log^{1-k}(u)} = 0$$

If $k - 1 > 0$ we can apply L'hospital's rule k-1 more times until the exponent above $log(u)$ is smaller or equal 0. Then we can rewrite the expression as in the first case and get 0 in the limit. □

PROPOSITION 6. *It is not possible that:*

$$BayesRegret(T) \in \widetilde{O}\left(\frac{\sigma_2^2}{\sigma_1^2} N^{\frac{3}{2}} T^{-\frac{1}{2}}\right)$$

unless $BayesRegret(T) = 0$ for all T.

PROOF. Assume, by contradiction, that there exists T_0 for which BayesRegret$(T_0) > 0$ (from Proposition 4 we know that it cannot be smaller than 0) and the bound is possible. Denote $\alpha = \frac{\sigma_2^2}{\sigma_1^2} N^{\frac{3}{2}}$. We can use the definitions to get:

$$\exists T' \geq 1, c \geq 0, k \geq 0 :$$
$$\forall T \geq T' : \text{BayesRegret}(T) \leq c * \alpha T^{-\frac{1}{2}} log^k(\alpha T^{-\frac{1}{2}}) \tag{7}$$

From Proposition 5 we know that right-hand side of the inequality approaches 0, so we know from the definition of limit that for $\frac{\text{BayesRegret}(T_0)}{2}$:

$$\exists T'' \geq 1 : \forall T \geq T'' :$$

$$|c * \alpha T^{-\frac{1}{2}} log^k(\alpha T^{-\frac{1}{2}}) - 0| \leq \frac{\text{BayesRegret}(T_0)}{2} \tag{8}$$

From the inequalities (7) and (8) we get that:

$$\forall T \geq max(T', T'', T_0) :$$

$$\text{BayesRegret}(T) \leq c * \alpha T^{-\frac{1}{2}} log^k(\alpha T^{-\frac{1}{2}}) \leq |c * \alpha T^{-\frac{1}{2}} log^k(\alpha T^{-\frac{1}{2}})| \leq \frac{\text{BayesRegret}(T_0)}{2} < \text{BayesRegret}(T_0)$$

We obtained that for $T > T_0$:

$$\text{BayesRegret}(T) < \text{BayesRegret}(T_0)$$

which contradicts the Proposition 4. □

Proposition 7. *In case of the model formulation presented in section Assumptions:*

$$BayesRegret(T_1) > 0$$

Proof.

$$\text{BayesRegret}(T_1) = \mathbb{E}\left[\mathbb{E}\left[\mathbb{E}[\max_{i=1,\dots,N} \gamma_i - r_{i_1,1}|\theta, \gamma_1, \dots\gamma_N]\right]\middle|\theta\right]$$

Calculating the inner expected value we get:

$$\mathbb{E}[\max_{i=1,\dots,N} \gamma_i - r_{i_1,1}|\theta, \gamma_1, \dots\gamma_N] = \max_{i=1,\dots,N} \gamma_i - \sum_{i=1,\dots,N} \mathbb{E}[r_i|i_1 = i, \theta, \gamma_1, \dots\gamma_N] * \mathbb{P}(i_1 = i|\theta, \gamma_1, \dots\gamma_N)$$

For $t = 1$ distributions used by the LSPS algorithm are the same for all arms, so: $\mathbb{P}(i_1 = i|\theta, \gamma_1, \dots\gamma_N) = \frac{1}{N}$. Also $r_i|\gamma_i \sim \mathcal{N}(\gamma_i, \sigma_1^2)$. Using these facts we can write (6.1) as:

$$\max_{i=1,\dots,N} \gamma_i - \frac{1}{N} \sum_{i=1,\dots,N} \gamma_i$$

Value of this expression is always greater or equal zero. It is equal 0 if all γ_i have the same value. They are, conditionally on θ, independent and normally distributed, so probability of this event is 0 and we can infer that:

$$\mathbb{E}\left[\max_{i=1,\dots,N} \gamma_i - \frac{1}{N} \sum_{i=1,\dots,N} \gamma_i \middle|\theta\right] > 0$$

From this it follows that also the most outer $\mathbb{E}$ is greater than 0. □

REFERENCES

[1] Yasin Abbasi-Yadkori, Dávid Pál, and Csaba Szepesvári. 2011. Improved algorithms for linear stochastic bandits. In *Advances in Neural Information Processing Systems*. 2312–2320.

[2] Milton Abramowitz and Irene A Stegun. 1964. Handbook of mathematical functions, Natl. *Bur. Stand. Appl. Math. Ser* 55 (1964), 1046.

[3] Rajeev Agrawal. 1995. Sample mean based index policies with O (log n) regret for the multi-armed bandit problem. *Advances in Applied Probability* (1995), 1054–1078.

[4] Shipra Agrawal and Navin Goyal. 2013. Thompson sampling for contextual bandits with linear payoffs. In *International Conference on Machine Learning*. 127–135.

[5] Shipra Agrawal and Navin Goyal. 2017. Near-optimal regret bounds for thompson sampling. *Journal of the ACM (JACM)* 64, 5 (2017), 1–24.

[6] Robin Allesiardo, Raphaël Féraud, and Djallel Bouneffouf. 2014. A neural networks committee for the contextual bandit problem. In *International Conference on Neural Information Processing*. Springer, 374–381.

[7] Peter Auer. 2002. Using confidence bounds for exploitation-exploration trade-offs. *Journal of Machine Learning Research* 3, Nov (2002), 397–422.

[8] Peter Auer, Nicolo Cesa-Bianchi, and Paul Fischer. 2002. Finite-time analysis of the multiarmed bandit problem. *Machine learning* 47, 2-3 (2002), 235–256.

[9] Olivier Cappé, Aurélien Garivier, Odalric-Ambrym Maillard, Rémi Munos, Gilles Stoltz, et al. 2013. Kullback–leibler upper confidence bounds for optimal sequential allocation. *The Annals of Statistics* 41, 3 (2013), 1516–1541.

[10] Olivier Chapelle and Lihong Li. 2011. An empirical evaluation of thompson sampling. In *Advances in neural information processing systems*. 2249–2257.

[11] Wei Chu, Lihong Li, Lev Reyzin, and Robert Schapire. 2011. Contextual bandits with linear payoff functions. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*. 208–214.

[12] Mark Collier and Hector Urdiales Llorens. 2018. Deep Contextual Multi-armed Bandits. *arXiv preprint arXiv:1807.09809* (2018).

[13] Varsha Dani, Thomas P Hayes, and Sham M Kakade. 2008. Stochastic linear optimization under bandit feedback. (2008).

[14] Adam N Elmachtoub, Ryan McNellis, Sechan Oh, and Marek Petrik. 2017. A practical method for solving contextual bandit problems using decision trees. *arXiv preprint arXiv:1706.04687* (2017).

[15] Raphaël Féraud, Robin Allesiardo, Tanguy Urvoy, and Fabrice Clérot. 2016. Random forest for the contextual bandit problem. In *Artificial Intelligence and Statistics*. 93–101.

[16] Sarah Filippi, Olivier Cappe, Aurélien Garivier, and Csaba Szepesvári. 2010. Parametric bandits: The generalized linear case. In *Advances in Neural Information Processing Systems*. 586–594.

[17] Dylan J Foster, Alekh Agarwal, Miroslav Dudík, Haipeng Luo, and Robert E Schapire. 2018. Practical contextual bandits with regression oracles. *arXiv preprint arXiv:1803.01088* (2018).

[18] Claudio Gentile, Shuai Li, Purushottam Kar, Alexandros Karatzoglou, Giovanni Zappella, and Evans Etrue. 2017. On context-dependent clustering of bandits. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*. JMLR. org, 1253–1262.

[19] Avishek Ghosh, Sayak Ray Chowdhury, and Aditya Gopalan. 2017. Misspecified linear bandits. In *Thirty-First AAAI Conference on Artificial Intelligence*.

[20] Thore Graepel, Joaquin Quinonero Candela, Thomas Borchert, and Ralf Herbrich. 2010. Web-scale bayesian click-through rate prediction for sponsored search advertising in microsoft's bing search engine. Omnipress.

[21] Ole-Christoffer Granmo. 2010. Solving two-armed Bernoulli bandit problems using a Bayesian learning automaton. *International Journal of Intelligent Computing and Cybernetics* (2010).

[22] Samarth Gupta, Gauri Joshi, and Osman Yağan. 2020. Correlated multi-armed bandits with a latent random source. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 3572–3576.

[23] 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*. 99–109.

[24] Emilie Kaufmann, Olivier Cappé, and Aurélien Garivier. 2012. On Bayesian upper confidence bounds for bandit problems. In *Artificial intelligence and statistics*. 592–600.

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

[26] Gi-Soo Kim and Myunghee Cho Paik. 2019. Doubly-Robust Lasso Bandit. In *Advances in Neural Information Processing Systems 32*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (Eds.). Curran Associates, Inc., 5877–5887. http://papers.nips.cc/paper/8822-doubly-robust-lasso-bandit.pdf

[27] Andreas Krause and Cheng S Ong. 2011. Contextual gaussian process bandit optimization. In *Advances in neural information processing systems*. 2447–2455.

[28] Akshay Krishnamurthy, Zhiwei Steven Wu, and Vasilis Syrgkanis. 2018. Semiparametric contextual bandits. *arXiv preprint arXiv:1803.04204* (2018).

[29] Branislav Kveton, Manzil Zaheer, Csaba Szepesvari, Lihong Li, Mohammad Ghavamzadeh, and Craig Boutilier. 2020. Randomized exploration in generalized linear bandits. In *International Conference on Artificial Intelligence and Statistics*. 2066–2076.

[30] Tze Leung Lai and Herbert Robbins. 1985. Asymptotically efficient adaptive allocation rules. *Advances in applied mathematics* 6, 1 (1985), 4–22.

[31] Lihong Li, Wei Chu, John Langford, and Robert E Schapire. 2010. A contextual-bandit approach to personalized news article recommendation. In *Proceedings of the 19th international conference on World wide web*. ACM, 661–670.

[32] Lihong Li, Yu Lu, and Dengyong Zhou. 2017. Provably optimal algorithms for generalized linear contextual bandits. *arXiv preprint arXiv:1703.00048* (2017).

[33] Shuai Li, Alexandros Karatzoglou, and Claudio Gentile. 2016. Collaborative filtering bandits. In *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval*. 539–548.

[34] Benedict C May and David S Leslie. 2011. Simulation studies in optimistic Bayesian sampling in contextual-bandit problems. *Statistics Group, Department of Mathematics, University of Bristol* 11, 02 (2011).

[35] Mingdong Ou, Nan Li, Cheng Yang, Shenghuo Zhu, and Rong Jin. 2019. Semi-parametric sampling for stochastic bandits with many arms. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 33. 7933–7940.

[36] Yi Peng, Miao Xie, Jiahao Liu, Xuying Meng, Nan Li, Cheng Yang, Tao Yao, and Rong Jin. 2019. A practical semi-parametric contextual bandit. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence*. AAAI Press, 3246–3252.

[37] Vianney Perchet, Philippe Rigollet, et al. 2013. The multi-armed bandit problem with covariates. *The Annals of Statistics* 41, 2 (2013), 693–721.

[38] Philippe Rigollet and Assaf Zeevi. 2010. Nonparametric bandits with covariates. *arXiv preprint arXiv:1003.1630* (2010).

[39] Paat Rusmevichientong and John N Tsitsiklis. 2010. Linearly parameterized bandits. *Mathematics of Operations Research* 35, 2 (2010), 395–411.

[40] Daniel Russo and Benjamin Van Roy. 2014. Learning to optimize via posterior sampling. *Mathematics of Operations Research* 39, 4 (2014), 1221–1243.

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

[42] Daniel J. 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. https://doi.org/10.1561/2200000070

[43] Steven L Scott. 2010. A modern Bayesian look at the multi-armed bandit. *Applied Stochastic Models in Business and Industry* 26, 6 (2010), 639–658.

[44] Aleksandrs Slivkins. 2011. Contextual bandits with similarity information. In *Proceedings of the 24th annual Conference On Learning Theory*. JMLR Workshop and Conference Proceedings, 679–702.

[45] Niranjan Srinivas, Andreas Krause, Sham M Kakade, and Matthias W Seeger. 2012. Information-theoretic regret bounds for gaussian process optimization in the bandit setting. *IEEE Transactions on Information Theory* 58, 5 (2012), 3250–3265.

[46] 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.

[47] Michal Valko, Nathaniel Korda, Rémi Munos, Ilias Flaounas, and Nelo Cristianini. 2013. Finite-time analysis of kernelised contextual bandits. *arXiv preprint arXiv:1309.6869* (2013).

[48] Qing Wang, Chunqiu Zeng, Wubai Zhou, Tao Li, S Sitharama Iyengar, Larisa Shwartz, and Genady Ya Grabarnik. 2018. Online interactive collaborative filtering using multi-armed bandit with dependent arms. *IEEE Transactions on Knowledge and Data Engineering* 31, 8 (2018), 1569–1580.