# Low-Rank Generalized Linear Bandit Problems

**Yangyi Lu**
University of Michigan
yylu@umich.edu

**Amirhossein Meisami**
Adobe Inc.
meisami@adobe.com

**Ambuj Tewari**
University of Michigan
tewaria@umich.edu

## Abstract

In a low-rank linear bandit problem, the expected reward of an action (represented by a matrix of size $d_1 \times d_2$) is the inner product between the action and an unknown low-rank matrix $\Theta^*$. We propose an algorithm based on a novel combination of online-to-confidence-set conversion (Abbasi-Yadkori et al., 2012) and the exponentially weighted average forecaster constructed by a covering of low-rank matrices. In $T$ rounds, our algorithm achieves $\widetilde{O}((d_1 + d_2)^{3/2}\sqrt{rT})$ regret that improves upon the standard linear bandit regret bound of $\widetilde{O}(d_1 d_2\sqrt{T})$ when the rank of $\Theta^*$: $r \ll \min\{d_1, d_2\}$. We also extend our algorithmic approach to the generalized linear setting to get an algorithm which enjoys a similar bound under regularity conditions on the link function. To get around the computational intractability of covering based approaches, we propose an efficient algorithm by extending the "Explore-Subspace-Then-Refine" algorithm of Jun et al. (2019). Our efficient algorithm achieves $\widetilde{O}((d_1 + d_2)^{3/2}\sqrt{rT})$ regret under a mild condition on the action set $\mathcal{X}$ and the $r$-th singular value of $\Theta^*$. Our upper bounds match the conjectured lower bound of Jun et al. (2019) for a subclass of low-rank linear bandit problems. Further, we show that existing lower bounds for the sparse linear bandit problem strongly suggest that our regret bounds are unimprovable. To complement our theoretical contributions, we also conduct experiments to demonstrate that our algorithm can greatly outperform the performance of the standard linear bandit approach when $\Theta^*$ is low-rank.

Proceedings of the 24th International Conference on Artificial Intelligence and Statistics (AISTATS) 2021, San Diego, California, USA. PMLR: Volume 130. 

## 1 INTRODUCTION

Low-rank models are widely used in various applications, such as matrix completion, computer vision, etc (Candès and Recht, 2009; Basri and Jacobs, 2003). We study low-rank (generalized) linear models in the bandit setting (Lai and Robbins, 1985). During the learning process, the agent adaptively pulls an arm (denoted as $X_t$) from a set of arms based on the past experience. At each pull, the agent observes a noisy reward corresponding to the arm pulled. Let $\Theta^* \in \mathbb{R}^{d_1 \times d_2}$ be an unknown low-rank matrix with rank $r \ll \min\{d_1, d_2\}$. The learner's goal is to maximize the total reward: $\sum_{t=1}^{T} \mu\left(\langle \Theta^*, X_t \rangle\right)$ where $T$ is the time horizon, $X_t \in \mathbb{R}^{d_1 \times d_2}$ is an action pulled at time $t$ that belongs to a pre-specified action set $\mathcal{X}$ and $\mu(\cdot)$ denotes a link function. Note that in the standard linear case the link function is identity.

Many practical applications can be framed in this low-rank bandit model, where the rank of arm features has no restriction. For traveling websites, the recommendation system needs to choose a flight-hotel bundle for the customer that can achieve high revenue. Often one has $m$ features of size $d_1$ for a flight $(x_1, \ldots, x_m \in \mathbb{R}^{d_1})$ and $m$ features of size $d_2$ for a hotel $(y_1, \ldots, y_m \in \mathbb{R}^{d_2})$. It is natural to form a $d_1 \times d_2$ matrix feature via outer products summation $\sum_{i=1}^{m} x_i y_i^T$ for each bundle, the rank of which can be any value in $\{0, 1, \ldots, \min\{m, d_1, d_2\}\}$. One can model the appeal of a bundle by a (generalized) linear function of the matrix feature $\sum_{i=1}^{m} x_i y_i^T$. In online advertising with image recommendation, the advertiser selects an image to display and the goal is to achieve the maximum clicking rate. The image is often stored as a $d_1 \times d_2$ matrix, and one can use a generalized linear model (GLM) with the link function being the logistic function to model the click rate (Richardson et al., 2007; McMahan et al., 2013). In all of these applications, one puts some capacity control on the underlying matrix linear coefficient $\Theta^*$ and a natural condition is $\Theta^*$ being low-rank. We note that the examples such as online dating and online shopping discussed in Jun et al. (2019) can also be formulated as our model.

In this paper, we measure the quality of an algorithm in terms of its cumulative regret[1]. A naive approach is to ignore the low-rank structure and directly apply the standard (generalized) linear bandit algorithms (Abbasi-Yadkori et al., 2011; Filippi et al., 2010). These approaches suffer $\widetilde{O}(d_1 d_2 \sqrt{T})$ regret.[2] However, in practice, $d_1 d_2$ can be huge. Then a natural question is:

*Can we utilize the low-rank structure of $\Theta^*$ to achieve $o(d_1 d_2 \sqrt{T})$ regret?*

Jun et al. (2019) studied a *subclass* of our problem, where the actions are *rank one* matrices. They proposed an algorithm that achieves $\widetilde{O}((d_1+d_2)^{3/2}\sqrt{rT})$ regret under additional incoherence and singular value assumptions of an augmented matrix defined via the arm set and $\Theta^*$ and a singular value assumption of $\Theta^*$. They also provided strong evidence that their bound is unimprovable.

We summarize our contributions below.

1. We propose Low Rank Linear Bandit with Online Computation algorithm (LowLOC) for the low-rank linear bandit problem, that achieves $\widetilde{O}((d_1+d_2)^{3/2}\sqrt{rT})$ regret. Notably, comparing with the result in Jun et al. (2019), our result
   - applies to more general action sets which can contain high-rank matrices and
   - does not require the incoherence and bounded eigenvalue assumption of the augmented matrix mentioned in the previous paragraph.

   Our regret bound also matches with their conjectured lower bound. For LowLOC, we first design a novel online predictor which uses an *exponentially weighted average forecaster* on a covering of low-rank matrices to solve the online low-rank linear prediction problem with $O((d_1+d_2)r\log T)$ regret. We then plug in our online predictor to the online-to-confidence-set conversion framework proposed by Abbasi-Yadkori et al. (2012) to construct a confidence set of $\Theta^*$ in our bandit setting, and at every round we choose the action optimistically.
2. We further propose Low Rank Generalized Linear Bandit with Online Computation algorithm (LowGLOC) for the generalized linear setting that also achieves $\widetilde{O}((d_1+d_2)^{3/2}\sqrt{rT})$ regret. LowGLOC is similar to LowLOC but here we need to design a new online-to-confidence-set conversion method, which can be of independent interest.
3. LowLOC and LowGLOC enjoy good regret but are unfortunately not efficiently implementable. To overcome this issue, we provide an efficient algorithm Low-Rank-Explore-Subspace-Then-Refine (LowESTR) for the linear setting, inspired by the ESTR algorithm proposed by Jun et al. (2019). We show that under a mild assumption on action set $\mathcal{X}$, LowESTR achieves $\widetilde{O}((d_1+d_2)^{3/2}\sqrt{rT}/\omega_r)$ regret, where $\omega_r > 0$ is a lower bound for the $r$-th singular value of $\Theta^*$. Comparing with ESTR, LowESTR does not need the incoherence and the eigenvalue assumption of the augmented matrix while the assumptions on the action set of the two algorithms are different. We also provide empirical evaluations to demonstrate the effectiveness of LowESTR.

[1] See Section 3 for the definition.

[2] $\widetilde{O}$ omits poly-logarithmic factors of $d_1, d_2, r, T$.

## 2 RELATED WORK

Our work is inspired by Jun et al. (2019) where they model the reward as $x_t^\top \Theta^* z_t$. $x_t \in \mathcal{X} \subset \mathbb{R}^{d_1}$ is a left arm and $z_t \in \mathcal{Z} \subset \mathbb{R}^{d_2}$ is a right arm ($\mathcal{X}$ and $\mathcal{Z}$ are left and right arm sets, repsectively). Note this model is a special case of our low-rank linear bandit model because one can write $x_t^\top \Theta^* z_t = \langle \Theta^*, x_t z_t^\top \rangle$ and define the arm set as $\mathcal{X}\mathcal{Z}^\top$. Their ESTR algorithm enjoys $O((d_1+d_2)^{3/2}\sqrt{rT}/\omega_r)$ regret bound under the assumptions: 1) an augmented matrix $K^* = X\Theta^* Z^\top$ is incoherent (Keshavan et al., 2010) and has a finite condition number, where $X \in \mathbb{R}^{d_1 \times d_1}$ is constructed by $d_1$ arms from $\mathcal{X}$ that maximizes $\|X^{-1}\|_2$ and $Z \in \mathbb{R}^{d_2 \times d_2}$ is constructed by $d_2$ arms from $\mathcal{Z}$ that maximizes $\|Z^{-1}\|_2$, and 2) $\|X^{-1}\|_2$ and $\|Z^{-1}\|_2$ are upper bounded by a constant. Their algorithm requires explicitly finding $X$ and $Z$, which is in general NP-hard, even though they also proposed heuristics to speed up this step. Comparing with ESTR, our LowLOC and LowGLOC algorithm are also not computationally efficient, but they both apply to richer action sets (matrices of any rank) without assumptions on $K^*$, $X$ and $Z$ and their regret bound does not depend on $\omega_r$. Our LowESTR algorithm is computationally efficient if the action set admits a nice exploration distribution (see details in Section 6). LowESTR achieves $O\left((d_1+d_2)^{3/2}\sqrt{rT}/\omega_r\right)$ regret bound but it does not require assumptions on $K^*$, $X$ and $Z$ as well.

Katariya et al. (2017b) and Kveton et al. (2017) also studied rank-1 and low-rank bandit problems. They assume there is an underlying expected reward matrix $\bar{R}$, at each time the learner picks an element on $(i_t, j_t)$ position and receives a noisy reward. It can be viewed as a special case of bilinear bandit with one-hot vectors as left and right arms. Katariya et al. (2017b) is further extended by Katariya et al. (2017a) that uses KL based confidence intervals to achieve a tighter regret bound. Our problem is more general comparing to these works. Johnson et al. (2016) considered the

same setting as ours, but their method relies on the knowledge of many parameters that depend on the unknown $\Theta^*$ and in particular only works for continuous arm set.

There are other works that utilize the low-rank structure in different model settings. For example, Gopalan et al. (2016) studied low rank bandits with latent structures using robust tensor power method. Lale et al. (2019) imposed low-rank assumptions on the feature vectors to reduce the effective dimension. These work all utilize the low-rank structure to achieve better regret bound than standard approaches that do not take the low-rank structure into account.

## 3 PRELIMINARIES

We formally define the problem and review relevant background in this section.

### 3.1 Low-rank Linear Bandit

Let $\mathcal{X} \subset \mathbb{R}^{d_1 \times d_2}$ be the arm space. In each round $t$, the learner chooses an arm $X_t \in \mathcal{X}$, and observes a noisy reward of a linear form:

$$y_t = \langle X_t, \Theta^* \rangle + \eta_t,$$

where $\Theta^* \in \mathbb{R}^{d_1 \times d_2}$ is an unknown parameter and $\eta_t$ is a 1-sub-Gaussian random variable. Denote the rank of $\Theta^*$ by $r$, we assume $r \ll \min\{d_1, d_2\}$. Let the $r$-th singular value of $\Theta^*$ is lower bounded by $\omega_r > 0$. We use $\langle A, B \rangle := \mathbf{trace}(A^T B)$ to denote the inner product between matrix $A$ and $B$. We follow the standard assumptions in linear bandits:

$$\|\Theta^*\|_F \leq 1 \text{ and } \|X\|_F \leq 1, \text{ for all } X \in \mathcal{X}.$$

In this low-rank linear bandit problem, the goal of the learner is to maximize the total reward $\sum_{t=1}^{T} \langle X_t, \Theta^* \rangle$, where $T$ is the time horizon. Clearly, with the knowledge of the unknown parameter $\Theta^*$, one should always select an action $X^* \in \text{argmax}_{X \in \mathcal{X}} \langle X, \Theta^* \rangle$. It is natural to evaluate the learner relative to the optimal strategy. The difference between the learner's total reward and the total reward of the optimal strategy is called *pseudo-regret* (Audibert et al., 2009):

$$R_T := \sum_{t=1}^{T} \langle X^* - X_t, \Theta^* \rangle.$$

For simplicity, we use the word regret instead of pseudo-regret for $R_T$.

### 3.2 Generalized Low-rank Linear Bandit

We also study the *generalized linear bandit model* of the following form: $\mathbb{E}[y_t | X_t, \Theta^*] = \mu(\langle X_t, \Theta^* \rangle)$ where $\mu(\cdot)$ is a link function. This framework builds on the well-known Generalized Linear Models (GLMs) and has been widely studied in many applications. For example, when rewards are binary-valued, a natural link function is the logistic function $\mu(x) = \exp(x)/(1 + \exp(x))$. For the generalized setting, we assume the reward given the action follows an exponential family distribution:

$$\mathbb{P}(y | z = \langle X, \Theta^* \rangle) = \exp\left(\frac{yz - m(z)}{\phi(\tau)} + h(y, \tau)\right), \tag{1}$$

where $\tau \in \mathbb{R}^+$ is a known scale parameter and $m, \phi$ and $h$ are some known functions. From basic calculation we get $m'(z) = \mathbb{E}[y|z] := \mu(z)$. We assume the above exponential family is a minimal representation, then $m(z)$ is ensured to be strictly convex (Wainwright and Jordan, 2008), and thus the negative log likelihood (NLL) loss $\ell(z, y) := -yz + m(z)$ is also strictly convex.

We make the following standard assumption on the link function $\mu(\cdot)$ (Jun et al., 2017).

**Assumption 1.** *There exist constants $L_\mu, c_\mu \geq 0, \kappa_\mu > 0$, such that the link function $\mu(\cdot)$ is $L_\mu-$Lipschitz on $[-1, 1]$, continously differentiable on $(-1, 1)$, $\inf_{z \in (-1,1)} \mu'(z) := \kappa_\mu$ and $|\mu(0)| \leq c_\mu$.*

One can write down the above reward model (1) in an equivalent way:

$$y_t = \mu(\langle X_t, \Theta^* \rangle) + \eta_t,$$

where $\eta_t$ is conditionally $R$-sub-Gaussian given $X_t$ and $\{(X_s, \eta_s)\}_{s=1}^{t-1}$. Using the form of $\mathbb{P}(y|z)$, Taylor expansion and the strictly convexity of $m(\cdot)$, one can show that $R = \sup_{z \in [-1,1]} \sqrt{\mu'(z)} \leq \sqrt{L_\mu}$ by the definition of the sub-Gaussian constant. An optimal arm is $X^* \in \text{argmax}_{X \in \mathcal{X}} \mu(\langle X, \Theta^* \rangle)$. The performance of an algorithm is again evaluated by cumulative regret:

$$R_T = \sum_{t=1}^{T} \mu(\langle X^*, \Theta^* \rangle) - \mu(\langle X_t, \Theta^* \rangle).$$

**Other notations.** We use $O$ and $\Omega$ for the standard Big O and Big Omega notations. $\widetilde{O}$ and $\widetilde{\Omega}$ ignore the poly-logarithmic factors of $d_1, d_2, r, T$. $f(x) \asymp g(x)$ indicates $f$ and $g$ are of the same order ignoring the poly-logarithmic factors of $d_1, d_2, r, T$. For any set $\mathcal{S}$, we use $|\mathcal{S}|$ to denote its cardinality.

## 4 LOW-RANK LINEAR BANDIT WITH ONLINE COMPUTATION

We first present our algorithm, LowLOC (Algorithm 1) for low-rank linear bandit problems. Before diving into details, we summarize our results as follows:

**Algorithm 1** Low-Rank Linear Bandit with Online Computation (LowLOC)

**Input:** arm set: $\mathcal{X}$, horizon: $T$, $\frac{1}{T}$-net for $S_r$: $\bar{S}_r(\frac{1}{T})$, failure rate $\delta$, EW constant $\eta \asymp \frac{1}{\log(T/\delta)}$.
Initial confidence set $C_0 = \{\Theta \in \mathbb{R}^{d_1 \times d_2} : \|\Theta\|_F^2 \leq 1\}$.
**for** $t = 1, \ldots, T$ **do**
  $(X_t, \widetilde{\Theta}_t) := \text{argmax}_{(X,\Theta) \in \mathcal{X} \times C_{t-1}} \langle X, \Theta \rangle$.
  Pull arm $X_t$ and receive reward $y_t$.
  Compute EW predictor $\hat{y}_t = \frac{\sum_{i=1}^{|\bar{S}_r(\frac{1}{T})|} e^{-\eta L_{i,t-1}} f_{\Theta_i,t}}{\sum_{j=1}^{|\bar{S}_r(\frac{1}{T})|} e^{-\eta L_{j,t-1}}}$, where $f_{\Theta_i,t} \triangleq \langle X_t, \Theta_i \rangle$ for $\Theta_i \in \bar{S}_r(\frac{1}{T})$.
  Update losses $L_{i,t} = \sum_{s=1}^{t} (y_s - f_{\Theta_i,s})^2$, for $i = 1, \ldots, |\bar{S}_r(\frac{1}{T})|$.
  Update $\mathcal{C}_t$ according to Equation (2), where $B_t$ is defined in Lemma 2.
**end for**

**Theorem 1** (Regret of LowLOC (Algorithm 1)). *For $\forall \delta \in (0, 0.25]$, with probability at least $1 - \delta$, Algorithm 1 achieves regret:*

$$R_T = \widetilde{O}\left((d_1 + d_2)^{3/2} \sqrt{rT} \sqrt{\log\left(\frac{1}{\delta}\right)}\right).$$

Note that LowLOC achieves the desired goal of outperforming the standard linear bandit approach with $\widetilde{O}(d_1 d_2 \sqrt{T})$ regret. Furthermore, this bound does not depend on any other problem-dependent parameters such as least singular value of $\Theta^*$ and does not require any other assumption which appeared in Jun et al. (2019). In the following sub-sections, we explain details of our algorithm design choices.

### 4.1 OFU and Online-to-confidence-set Conversion

This algorithm follows the standard Optimism in the Face of Uncertainty (OFU) principle. We maintain a confidence set $\mathcal{C}_t$ at every round that contains the true parameter $\Theta^*$ with high probability and we choose the action $X_t$ according to

$$(X_t, \widetilde{\Theta}_t) = \text{argmax}_{(X,\Theta) \in \mathcal{X} \times \mathcal{C}_{t-1}} \langle X, \Theta \rangle.$$

Typically, the faster $\mathcal{C}_t$ shrinks, the lower regret we have. The main diffculty is to construct $\mathcal{C}_t$ that leverages the low-rank structure so that we only have $\widetilde{O}((d_1 + d_2)^{3/2} \sqrt{rT})$ regret. Our starting point is to use the online-to-confidence-set conversion framework proposed by Abbasi-Yadkori et al. (2012) who builds the confidence set based on an online predictor. At each round, an online predictor receives $X_t$, predicts $\hat{y}_t$, based on historical data $\{(X_s, y_s)\}_{s=1}^{t-1}$, observes the true value $y_t$ and suffers a loss $\ell_t(\hat{y}_t) \triangleq (y_t - \hat{y}_t)^2$. The performance of this online predictor is measured by comparing its cumulative loss to the cumulative loss of a fixed linear predictor using coefficient $\Theta$:

$$\rho_t(\Theta) = \sum_{s=1}^{t} \ell_s(\hat{y}_s) - \ell_s(\langle \Theta, X_s \rangle).$$

The key idea of online-to-confidence-set conversion (adapted to our low-rank setting) is that if one can guarantee $\sup_{\|\Theta\|_F \leq 1, \text{rank}(\Theta) \leq r} \rho_t(\Theta) \leq B_t$ for some non-decreasing sequence $\{B_t\}_{t=1}^{T}$, we can use this information to construct the confidence interval for $\Theta^*$ as:

$$\mathcal{C}_t = \{\Theta \in \mathbb{R}^{d_1 \times d_2} : \|\Theta\|_F^2 + \sum_{s=1}^{t} (\hat{y}_s - \langle \Theta, X_s \rangle)^2 \leq 1 + \beta_t(\delta)\}, \quad (2)$$

where $\beta_t(\delta) = 1 + 2B_t + 32 \log\left(\left(\sqrt{8} + \sqrt{1 + B_t}\right)/\delta\right)$ and $\delta$ is the failure probability.

Lemma 8 in appendix guarantees that $\Theta^*$ is contained in $\cap_{t \geq 1} \mathcal{C}_t$ with high probability and Lemma 9 further guarantees the overall regret

$$R_T = \widetilde{O}(\sqrt{d_1 d_2 \beta_{T-1}(\delta) T}) = \widetilde{O}\left((d_1 + d_2)\sqrt{B_{T-1} T}\right). \quad (3)$$

Therefore, the problem to achieve the $\widetilde{O}((d_1 + d_2)^{3/2} \sqrt{rT})$ regret bound reduces to designing an online predictor which guarantees $\sup_{\|\Theta\|_F \leq 1, \text{rank}(\Theta) \leq r} \rho_t(\Theta) \leq B_t$ and $B_t = \widetilde{O}((d_1 + d_2) r)$. To achieve this rate, the key is to leverage the low-rank structure of $\Theta^*$.

### 4.2 Online Low Rank Linear Prediction

We adopt the classical *exponentially weighted average forecaster* (EW) framework (Cesa-Bianchi and Lugosi 2006) which uses $N$ experts to predict $\hat{y}_t$ with the following formula

$$\widehat{y}_t = \frac{\sum_{i=1}^{N} e^{-\eta L_{i,t-1}} f_{i,t}}{\sum_{j=1}^{N} e^{-\eta L_{j,t-1}}}. \quad (4)$$

In above, $f_i$ denotes the $i$-th expert that makes a prediction $f_{i,t}$ at time $t$, $L_{i,t-1} \triangleq \sum_{s=1}^{t-1} \ell_s(f_i(X_t))$ is the cumulative loss incurred by expert $i$, and $\eta$ is a tuning parameter. By choosing $\eta$ carefully, one can guarantee that this predictor achieves $O(\log N \log(T/\delta))$ regret comparing with the best expert among the expert set.

See backgrounds on the construction of EW in Section G and Proposition 3.1 in Cesa-Bianchi and Lugosi (2006).

In our setting, an expert can be viewed as a matrix $\Theta$ that satisfies $\|\Theta\|_F \leq 1$ and $\text{rank}(\Theta) \leq r$, and makes prediction according to $f_{\Theta,t} \triangleq \langle \Theta, X_t \rangle$. There are infinitely many such experts and therefore we cannot directly use EW which requires finite number of experts. Our main idea is to construct $N$ experts which guarantees $\log N$ is small and these $N$ experts can represent the original expert set $S_r \triangleq \{\Theta \in \mathbb{R}^{d_1 \times d_2} : \|\Theta\|_F \leq 1, \text{rank}(\Theta) \leq r\}$ well, and then apply EW using these $N$ experts. We construct an $\varepsilon$-net $\bar{S}_r(\varepsilon)$, i.e., for any $\Theta \in S_r$, there exists a $\bar{\Theta} \in \bar{S}_r(\varepsilon)$, such that $\left\|\Theta - \bar{\Theta}\right\|_F \leq \epsilon$. We further prove that $|\bar{S}_r(\varepsilon)| \leq (9/\varepsilon)^{(d_1+d_2+1)r}$ in Lemma 7, so the number of experts $N$ in Equation (4) is at most $(9T)^{(d_1+d_2+1)r}$ if we set $\varepsilon = 1/T$.

The following lemma summarizes the performance of this online predictor.

**Lemma 2** (Regret of EW under Squared Loss)**.** *Let* $\eta = \frac{1}{2(2+\sqrt{2\log(2T/\delta)})^2}$ *in EW forecaster* (4)*. Then, for any* $0 < \delta < 0.25$*, with probability at least* $1-\delta$*, we have*

$$\sup_{\|\Theta\|_F \leq 1, rank(\Theta) \leq r} \rho_T(\Theta) = \widetilde{O}\left((d_1+d_2) r \log\left(\frac{1}{\delta}\right)\right).$$

To obtain Theorem 1, one just needs to plug Lemma 2 into Equation (3) by defining $B_T$ as $\sup_{\|\Theta\|_F \leq 1, \text{rank}(\Theta) \leq r} \rho_T(\Theta)$.

## 5 LOW-RANK GENERALIZED LINEAR BANDIT

We also study the low-rank generalized linear bandit setting. The main structure of our algorithm LowGLOC (Algorithm 2) is similar to LowLOC, so we focus on the key differences in this section.

We still use EW to perform online predictions, but instead of the squared loss, we use negative log likelihood (NLL) loss $\ell_s(\hat{y}_s) = -\hat{y}_s y_s + m(\hat{y}_s)$ to construct the forecaster in Equation (4), where $m(\cdot)$ is as defined in Section 3. Therefore, the performance of EW using NLL loss relative to a fixed linear predictor $\Theta$ is measured by:

$$\rho_T^{\text{GLB}}(\Theta) = \sum_{t=1}^{T} -\hat{y}_t y_t + m(\hat{y}_t) - \sum_{t=1}^{T} -\langle \Theta, X_t \rangle y_t + m(\langle \Theta, X_t \rangle).$$

**Algorithm 2** Low-rank Generalized Linear Bandit with Online Computation (LowGLOC)

**Input:** arm set: $\mathcal{X}$, horizon: $T$, $\frac{1}{T}$-net for $S_r$: $\bar{S}_r(\frac{1}{T})$, failure rate $\delta$, EW constant $\eta \asymp \frac{1}{\log(T/\delta)}$, function $m(\cdot)$ in the generalized linear model.
Initial confidence set $C_0 = \{\Theta \in \mathbb{R}^{d_1 \times d_2} : \|\Theta\|_F^2 \leq 1\}$.
**for** $t = 1, \ldots, T$ **do**
  $(X_t, \widetilde{\Theta}_t) := \text{argmax}_{(X,\Theta) \in \mathcal{X} \times C_{t-1}} \langle X, \Theta \rangle$.
  Pull arm $X_t$ and receive reward $y_t$.
  Compute EW predictor $\hat{y}_t = \frac{\sum_{i=1}^{|\bar{S}_r(\frac{1}{T})|} e^{-\eta L_{i,t-1}} f_{\Theta_i,t}}{\sum_{j=1}^{|\bar{S}_r(\frac{1}{T})|} e^{-\eta L_{j,t-1}}}$, where $f_{\Theta_i,t} \triangleq \langle X_t, \Theta_i \rangle$ for $\Theta_i \in \bar{S}_r(\frac{1}{T})$.
  Update losses $L_{i,t} = \sum_{s=1}^{t} -f_{\Theta_i,s} y_s + m(f_{\Theta_i,s})$, for $i = 1, \ldots, |\bar{S}_r(\frac{1}{T})|$.
  Update $\mathcal{C}_t$ according to Equation (5), where $B_t^{\text{GLB}}$ is as defined in Lemma 14
**end for**

If there exists a non-decreasing sequence $\{B_t^{\text{GLB}}\}_{t=1}^T$ such that $\sup_{\|\Theta\|_F \leq 1, \text{rank}(\Theta) \leq r} \rho_t^{\text{GLB}}(\Theta) \leq B_t^{\text{GLB}}$, we construct $\mathcal{C}_t^{\text{GLB}}$ in the following way:

$$\mathcal{C}_t^{\text{GLB}} = \{\Theta \in \mathbb{R}^{d_1 \times d_2} : \|\Theta\|_F^2 + \sum_{s=1}^{t} (\hat{y}_s - \langle \Theta^*, X_s \rangle)^2 \leq \beta_t^{\text{GLB}}(\delta)\}, \quad (5)$$

where

$$\beta_t^{\text{GLB}}(\delta) = 2 + \frac{4}{\kappa_\mu} B_t^{\text{GLB}} + \frac{32 L_\mu}{\kappa_\mu^2} \log\left(\left(\sqrt{L_\mu}\sqrt{\frac{8}{\kappa_\mu^2}} + \sqrt{\frac{2}{\kappa_\mu} B_t^{\text{GLB}} + 1}\right)\frac{1}{\delta}\right)$$

and $\delta$ is the failure probability.

Lemma 12 guarantees that the true parameter $\Theta^*$ is contained in $\cap_{t \geq 1} \mathcal{C}_t^{\text{GLB}}$ with high probability. Lemma 13 further guarantees that the overall regret of LowGLOC satisfies

$$R_T = \widetilde{O}(\sqrt{d_1 d_2 \beta_{T-1}^{\text{GLB}}(\delta) T}) = \widetilde{O}((d_1+d_2)\sqrt{B_T^{\text{GLB}} T/\kappa_\mu}).$$

Following the online-to-confidence-set conversion idea as used in LowLOC, we prove that

$$B_T^{\text{GLB}} = O\left(\frac{L_\mu^2 + c_\mu^2}{\kappa_\mu}(d_1+d_2) r \log T \log\left(\frac{T}{\delta}\right)\right)$$

in Lemma 14

We next present the regret of LowGLOC in the following theorem, which can be easily achieved by plugging Lemma 14 into Lemma 13 as described in above paragraph.

**Theorem 3** (Regret of LowGLOC). *For $\forall \delta \in (0, 0.25]$, with probability at least $1-\delta$, Algorithm 2 achieves regret:*

$$R_T = \widetilde{O}\left((d_1+d_2)^{3/2}\sqrt{\frac{L_\mu^2+c_\mu^2}{\kappa_\mu^2} rT \log\left(\frac{1}{\delta}\right)}\right).$$

To the best of our knowledge, this is the first algorithm that achieves $o(d_1 d_2\sqrt{T})$ regret bound for low-rank GLM bandits.

# 6 EFFICIENT ALGORITHM FOR THE LINEAR CASE

At every round, LowLOC and LowGLOC need to calculate exponentially weighted predictions, which involves calculating weights of the covering of low-rank matrices. These approaches have high computation complexity even though their regret is ideal. In this section, we propose a computationally efficient method LowESTR (Algorithm 3) that also achieves $\widetilde{O}((d_1+d_2)^{3/2}\sqrt{rT})$ regret under mild assumptions on the action set $\mathcal{X}$ as follows.

**Assumption 2.** *There exists a sampling distribution $D$ over $\mathcal{X}$ with covariance matrix $\Sigma$, such that $\lambda_{min}(\Sigma) \asymp \frac{1}{d_1 d_2}$ and $D$ is sub-Gaussian with parameter $\sigma^2 \asymp \frac{1}{d_1 d_2}$. (see Definition 1 in Section C for the definition of sub-Gaussian random matrices.)*

This assumption is easily satisfied in many arm sets. To guarantee the existence of above sampling distribution $D$, we only need the convex hull of a subset of arms $\mathcal{X}_{sub} \subset \mathcal{X}$ contains a ball with radius $R \le 1$, which does not scale with $d_1$ or $d_2$. For example, if $\mathcal{X}$ is the Euclidean unit ball/sphere in $\mathbb{R}^{d_1 \times d_2}$, we can simply set $D$ to be the uniform distribution over $\mathcal{X}$. Notably, different choices of $D$ satisfying Assumption 2 do not affect the overall regret.

We extend the two-stage procedure "Explore Subspace Then Refine (ESTR)" proposed by Jun et al. (2019). In stage 1, ESTR estimates the row and column subspaces of $\Theta^*$. In stage 2, ESTR transforms the original problem into a $d_1 d_2$-dimensional linear bandit problem and invokes LowOFUL algorithm (Algorithm 4) (Jun et al., 2019), which leverages the estimated row/column subspaces of $\Theta^*$.

## 6.1 Description for LowESTR

LowESTR also proceeds with the two-stage framework as ESTR, but we use different estimation method in stage 1.

**Stage 1.** We are inspired by a line of work on low-rank matrices recovery using nuclear-norm penalty with squared loss (Wainwright, 2019). The learner pulls arm $X_t \in \mathcal{X}$ according to distribution $D$ and observes the reward $y_t$ up to a horizon $T_1$, then uses $\{X_t, y_t\}_{t=1}^{T_1}$ to solve a nuclear-norm penalized least square problem in (6) and receives an estimated $\widehat{\Theta}$ for $\Theta^*$. Notably, instead of invoking an NP-hard problem in stage 1 as ESTR, the optimization problem (6) in LowESTR is convex and thus can be solved easily using standard gradient based methods. Assumption 2 guarantees $\left\|\widehat{\Theta} - \Theta^*\right\|_F^2 \asymp \frac{(d_1+d_2)^3 r}{T_1}$ in Theorem 16 (Section E). We get the estimated row/column subspaces of $\Theta^*$ simply by running an SVD step.

**Stage 2.** In stage 2, we apply LowOFUL algorithm (Algorithm 4) proposed by Jun et al. (2019) in our setting. The key idea is reducing the problem to linear bandit and utilizing the estimated subspaces in the standard linear bandit method OFUL (Abbasi-Yadkori et al., 2011).

We now present the overall regret of Algorithm 3.

**Theorem 4** (Regret of LowESTR for Low Rank Bandit). *Suppose we run LowESTR in stage 1 with $T_1 \asymp (d_1+d_2)^{3/2}\sqrt{rT}\frac{1}{\omega_r}$ and $\lambda_{T_1}^2 \asymp \frac{1}{T_1 \min\{d_1,d_2\}}$. We invoke LowOFUL (Algorithm 4) in stage 2 with $k = r(d_1+d_2-r)$, $\lambda_\perp = \frac{T_2}{k\log(1+T_2/\lambda)}$, $B = 1$, $B_\perp = \gamma(T_1)$, and the rotated arm sets $\mathcal{X}'_{vec}$ defined in Algorithm 3, the overall regret of LowESTR is, with prob at least $1-2\delta$,*

$$R_T = \widetilde{O}\left((d_1+d_2)^{3/2}\sqrt{rT}\frac{1}{\omega_r}\right).$$

We believe that this "Explore-Subspace-Then-Refine" framework can also be extended to the generalized linear setting. In stage 1, an M-estimator that minimizes the negative log-likelihood plus nuclear norm penalty (Fan et al., 2019) can be used instead, while in stage 2, one can revise a standard generalized linear bandit algorithm such as GLM-UCB (Filippi et al., 2010) by leveraging the low-rank knowledge in the same way as LowOFUL. We leave this extension for future work.

## 6.2 Computational Complexity

Before we end this section, we note that the computational complexity of LowESTR is polynomial in the

**Algorithm 3** Low Rank Explore Subspace Then Refine (LowESTR)

**Input:** arm set $\mathcal{X}$, time horizon $T$, exploration length $T_1$, rank $r$ of $\Theta^*$, spectral bound $\omega_r$ of $\Theta^*$, sampling distribution for stage 1: $D$; parameters for LowOFUL in stage 2: $B, B_\perp, \lambda, \lambda_\perp$.

**Stage 1: Explore the Low Rank Subspace**

Pull $X_t \in \mathcal{X}$ according to distribution $D$ and observe reward $Y_t$, for $t = 1, \ldots, T_1$.

Solve $\widehat{\Theta}$ using the problem below:

$$\widehat{\Theta} = \text{argmin}_{\Theta \in \mathbb{R}^{d_1 \times d_2}} \frac{1}{2T_1} \sum_{t=1}^{T_1} (Y_t - \langle X_t, \Theta \rangle)^2 + \lambda_{T_1} \|\Theta\|_{\text{nuc}}. \tag{6}$$

Let $\widehat{\Theta} = U\widehat{S}V^T$ be the SVD of $\widehat{\Theta}$. Take the first $r$ columns of $U$ as $\widehat{U}$, the first $r$ rows of $V$ as $\widehat{V}$. Let $\widehat{U}_\perp$ and $\widehat{V}_\perp$ be orthonormal bases of the complementary subspaces of $\widehat{U}$ and $\widehat{V}$.

**Stage 2: Refine Standard Linear Bandit Algorithm**

Rotate the arm feature set: $\mathcal{X}' := \{[\widehat{U}\ \widehat{U}_\perp]^T X [\widehat{V}\ \widehat{V}_\perp] : X \in \mathcal{X}\}$.

Define a vectorized arm feature set so that the last $(d_1 - r)(d_2 - r)$ components are from the complementary subspaces:

$$\mathcal{X}'_{\text{vec}} := \{\text{vec}(X'_{1:r,1:r}); \text{vec}(X'_{r+1:d_1,1:r}); \text{vec}(X'_{1:r,r+1:d_2}); \text{vec}(X'_{r+1:d_1,r+1:d_2}) : X' \in \mathcal{X}'\}.$$

For $T_2 = T - T_1$ rounds, invoke LowOFUL (Algorithm 4) with arm set $\mathcal{X}'_{\text{vec}}$, the low dimension $k = (d_1 + d_2)r - r^2$ and $\gamma(T_1) \asymp \frac{(d_1+d_2)^3 r}{T_1 \omega_r^2}$, $B, B_\perp, \lambda, \lambda_\perp$.

**Algorithm 4** LowOFUL (Jun et al., 2019)

**Input:** $T, k$, arm set $\mathcal{A} \subset \mathbb{R}^{d_1 \times d_2}$, failure rate $\delta$ and positive constants $B, B_\perp, \lambda, \lambda_\perp$.

$\Lambda = \textbf{diag}(\lambda, \ldots, \lambda, \lambda_\perp, \ldots, \lambda_\perp)$, where $\lambda$ occupies the first $k$ diagonal entries.

**for** $t = 1, \ldots, T$ **do**

  Compute $a_t = \text{argmax}_{a \in \mathcal{A}} \max_{\theta \in \mathcal{C}_{t-1}} \langle \theta, a \rangle$.

  Pull arm $a_t$ and receive reward $y_t$.

  Update $C_t = \{\theta : \left\|\theta - \hat{\theta}\right\|_{V_t} \le \sqrt{\beta_t}\}$,

  where $\sqrt{\beta_t} = \sqrt{\log \frac{|V_t|}{|\Lambda|\delta^2}} + \sqrt{\lambda} B + \sqrt{\lambda_\perp} B_\perp$,

  $V_t = \Lambda + \sum_{s=1}^{t} a_t a_t^T$,

  $\hat{\theta}_t = (\Lambda + A^T A)^{-1} A^T \mathbf{y}$.

  (Here $A = [a_1^T; \ldots; a_t^T]$ and $\mathbf{y} := [y_1, \ldots, y_t]^T$).

**end for**

relevant quantities.

**Proposition 5** (Computational complexity of LowESTR)**.** *The computational complexity of LowESTR (Algorithm 3) is at most*

$$O\left(d_1 d_2 (d_1 + d_2)^3 r T / \omega_r^2 + d_1^2 d_2^2 T^2 + d_1^3 d_2^3 T\right).$$

In stage 1, we solve a convex optimization problem with unknown $\Theta \in \mathbb{R}^{d_1 \times d_2}$ using subgradient method, of which the complexity is $O(T_1 d_1 d_2 / \epsilon^2)$ ($\epsilon$ refers to the target accuracy). The complexity of the SVD step at the end of stage 1 is $O(d_1 d_2 \min\{d_1, d_2\})$.

In stage 2, LowOFUL algorithm (Algorithm 4) dominates the computational complexity. In iteration $t$ of LowOFUL, usually $a_t = \text{argmax}_{a \in \mathcal{A}} \max_{\theta \in \mathcal{C}_{t-1}} \langle \theta, a \rangle$ can be solved with an oracle in constant time, the complexity of least square estimation is $O(d_1^2 d_2^2 t + d_1^3 d_2^3)$ due to matrix multiplication and Cholesky factorization. Thus, in $T_2 \le T$ iterations, the computational complexity of stage 2 is at most $O(d_1^2 d_2^2 T^2 + d_1^3 d_2^3 T)$.

Combining the complexity results in two stages, taking the target accuracy $\epsilon = 1/\sqrt{T_1}$ and $T_1 = O\left((d_1 + d_2)^{3/2} \sqrt{rT} \frac{1}{\omega_r}\right)$ as stated in Theorem 4, the overall computational complexity in Proposition 5 is achieved.

## 7 LOWER BOUND FOR LOW-RANK LINEAR BANDIT

In this section, we discuss the regret lower bound of the low-rank linear bandit model. Suppose $d_1 = d_2 = d$, we first present a $\Omega(dr\sqrt{T})$ lower bound, which is a straightforward extension of the linear bandit lower bound (Lattimore and Szepesvári, 2018).

**Theorem 6** (Lower Bound)**.** *Assume $dr \le 2T$ and let $\mathcal{X} = \{X \in \mathbb{R}^{d \times d} : \|X\|_F \le 1\}$. Then $\exists \Theta \in \mathbb{R}^{d \times d}$, where $\|\Theta\|_F^2 \le \frac{d^2 r^2}{128T}$, rank$(\Theta) \le r$, s.t.*

$$\mathbb{E}\left[R_T(\Theta)\right] = \Omega(dr\sqrt{T}).$$

Above bound is tight when $r = d$ as it matches

with the standard $d^2$-dimensional linear bandit lower bound, but for small $r$, our upper bound is larger than the lower bound by a factor of $\sqrt{d/r}$.

Nevertheless, we conjecture that $\Omega(d^{3/2}\sqrt{rT})$ is the correct lower bound for small $r$. It is well-known that the regret lower bound for sparse linear bandit problem (dimension $d$, sparsity $s$) is $\Omega(\sqrt{sdT})$ (Lattimore and Szepesvári, 2018). Our low-rank linear bandit problem can be viewed as a $d^2$-dimensional linear bandit problem with $dr$ degrees of freedom in $\Theta^*$. Then, using the analogue of the degrees of freedom between sparse vectors and low-rank matrices, one can plug in $d^2$ for $d$ and $dr$ for $s$ in the sparse linear bandit regret lower bound and then achieve $\Omega(d^{3/2}\sqrt{rT})$ as our lower bound.

## 8 EXPERIMENTS

In this section, we compare the performance of OFUL and LowESTR to validate that it is crucial to utilize the low-rank structure.

We run our simulation with $d_1 = d_2 = 10, r = 1$ and $d_1 = d_2 = 10, r = 3$. In both settings, the true $\Theta^* \in \mathbb{R}^{d_1 \times d_2}$ is a diagonal matrix. For $r = 1$, we set $\text{diag}(\Theta^*) = (0.5, 0, \ldots, 0)$ while for $r = 3$, $\text{diag}(\Theta^*) = (0.5, 0.5, 0.5, 0, \ldots, 0)$. For arms in both settings, we draw 256 vectors from $N(0, I_{d_1 d_2})$ and standardize them by dividing their 2-norms, then we reshape all standardized $d_1 d_2$-dimensional vectors to $d_1 \times d_2$ matrices. We use these matrices as the arm set $\mathcal{X}$. For each arm $X \in \mathcal{X}$, the reward is generated by $y = \langle X, \Theta^* \rangle + \varepsilon$, where $\varepsilon \sim N(0, 0.01^2)$. We run both algorithms for $T = 3000$ rounds and repeat 100 times for each simulation setup to calculate the averaged regrets and their 1-standard deviation confidence intervals at every time step.

We leave the hyper-parameters of OFUL and LowESTR in the appendix (Section H). Regret comparison plots are displayed in Figure 1.

We observe that in both plots, LowESTR incurs less regret comparing to OFUL within several hundreds of time steps. Further, as we increase the rank from $r = 1$ to $r = 3$, the cumulative regret gap between the two approaches becomes smaller. This phenomenon is compatible with our theory.

Other than the comparisons between OFUL and LowESTR, we also conduct simulations to see the sensitivity of LowESTR to the eigenvalue parameter $\omega_r$. We observe that LowESTR indeed performs better as $\omega_r$ goes larger, which again matches with our theory. The detailed description and the plot for the sensitivity experiments are left to the appendix (Section H).

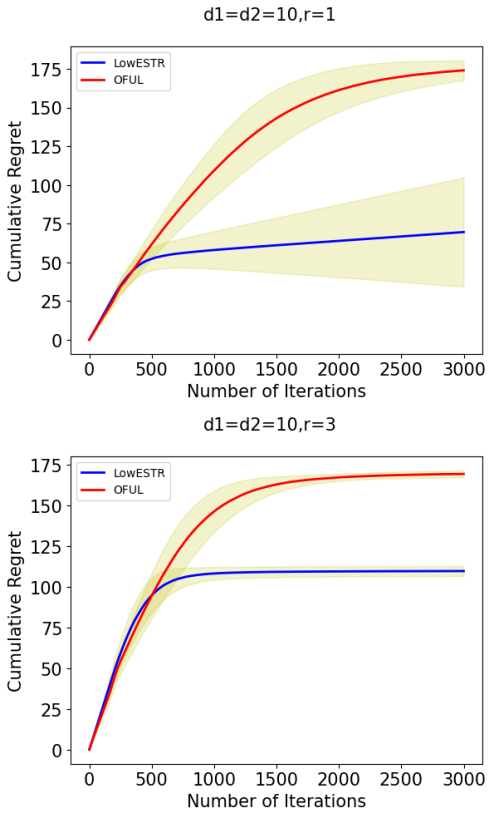


Figure 1: Regret comparison between OFUL and LowESTR for the two settings. We plot the averaged cumulative regret in red and blue curves, and 1-standard deviation for each method within the yellow shadow area.

## 9 CONCLUSION AND FUTURE WORK

In this paper, we studied the low-rank (generalized) linear bandit problem. We proposed LowLOC and LowGLOC algorithm for the linear and generalized linear setting, respectively. Both of them enjoy $\tilde{O}((d_1 + d_2)^{3/2}\sqrt{rT})$ regret. Further, our efficient algorithm LowESTR achieves $\tilde{O}((d_1 + d_2)^{3/2}\sqrt{rT}/\omega_r)$ regret under mild conditions on the action set.

There are several interesting directions we left for future work. First, building on some preliminary ideas in Section 6 about how to extend LowESTR to the generalized linear setting, it should be possible to obtain a similar regret bound under certain regularity conditions on the link function. Second, it will be interesting to investigate if one can design an efficient algorithm whose regret does not depend on $1/\omega_r$. Third, in Section 7, we argued that $\tilde{O}((d_1 + d_2)^{3/2}\sqrt{rT})$ should be a tight lower bound. It will be nice to formally prove this.

## ACKNOWLEDGEMENT

This work was supported in part by NSF CAREER grant IIS-1452099 and an Adobe Data Science Research Award.

### References

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

Abbasi-Yadkori, Y., Pal, D., and Szepesvari, C. (2012). Online-to-confidence-set conversions and application to sparse stochastic bandits. In *Artificial Intelligence and Statistics*, pages 1–9.

Audibert, J.-Y., Munos, R., and Szepesvári, C. (2009). Exploration–exploitation tradeoff using variance estimates in multi-armed bandits. *Theoretical Computer Science*, 410(19):1876–1902.

Basri, R. and Jacobs, D. W. (2003). Lambertian reflectance and linear subspaces. *IEEE transactions on pattern analysis and machine intelligence*, 25(2):218–233.

Candes, E. J. and Plan, Y. (2011). Tight oracle inequalities for low-rank matrix recovery from a minimal number of noisy random measurements. *IEEE Transactions on Information Theory*, 57(4):2342–2359.

Candès, E. J. and Recht, B. (2009). Exact matrix completion via convex optimization. *Foundations of Computational mathematics*, 9(6):717.

Cesa-Bianchi, N. and Lugosi, G. (2006). *Prediction, learning, and games*. Cambridge university press.

Fan, J., Gong, W., and Zhu, Z. (2019). Generalized high-dimensional trace regression via nuclear norm regularization. *Journal of econometrics*, 212(1):177–202.

Filippi, S., Cappe, O., Garivier, A., and Szepesvári, C. (2010). Parametric bandits: The generalized linear case. In *Advances in Neural Information Processing Systems*, pages 586–594.

Gopalan, A., Maillard, O.-A., and Zaki, M. (2016). Low-rank bandits with latent mixtures. *arXiv preprint arXiv:1609.01508*.

Johnson, N., Sivakumar, V., and Banerjee, A. (2016). Structured stochastic linear bandits. *arXiv preprint arXiv:1606.05693*.

Jun, K.-S., Bhargava, A., Nowak, R., and Willett, R. (2017). Scalable generalized linear bandits: Online computation and hashing. In *Advances in Neural Information Processing Systems*, pages 99–109.

Jun, K.-S., Willett, R., Wright, S., and Nowak, R. (2019). Bilinear bandits with low-rank structure. In *International Conference on Machine Learning*, pages 3163–3172.

Katariya, S., Kveton, B., Szepesvári, C., Vernade, C., and Wen, Z. (2017a). Bernoulli rank-1 bandits for click feedback. *arXiv preprint arXiv:1703.06513*.

Katariya, S., Kveton, B., Szepesvari, C., Vernade, C., and Wen, Z. (2017b). Stochastic rank-1 bandits. In *Artificial Intelligence and Statistics*, pages 392–401.

Keshavan, R. H., Montanari, A., and Oh, S. (2010). Matrix completion from noisy entries. *Journal of Machine Learning Research*, 11(Jul):2057–2078.

Kveton, B., Szepesvari, C., Rao, A., Wen, Z., Abbasi-Yadkori, Y., and Muthukrishnan, S. (2017). Stochastic low-rank bandits. *arXiv preprint arXiv:1712.04644*.

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

Lale, S., Azizzadenesheli, K., Anandkumar, A., and Hassibi, B. (2019). Stochastic linear bandits with hidden low rank structure. *arXiv preprint arXiv:1901.09490*.

Lattimore, T. and Szepesvári, C. (2018). Bandit algorithms. *preprint*.

Loh, P.-L. and Wainwright, M. J. (2011). High-dimensional regression with noisy and missing data: Provable guarantees with non-convexity. In *Advances in Neural Information Processing Systems*, pages 2726–2734.

McMahan, H. B., Holt, G., Sculley, D., Young, M., Ebner, D., Grady, J., Nie, L., Phillips, T., Davydov, E., Golovin, D., et al. (2013). Ad click prediction: a view from the trenches. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1222–1230.

Richardson, M., Dominowska, E., and Ragno, R. (2007). Predicting clicks: estimating the click-through rate for new ads. In *Proceedings of the 16th international conference on World Wide Web*, pages 521–530.

Wainwright, M. J. (2019). *High-dimensional statistics: A non-asymptotic viewpoint*, volume 48. Cambridge University Press.

Wainwright, M. J. and Jordan, M. I. (2008). Graphical models, exponential families, and variational inference. *Foundations and Trends® in Machine Learning*, 1(1-2):1–305.