

# Bandit Principal Component Analysis

**Wojciech Kotłowski**

*Poznan University of Technology, Poznań, Poland*

KOTLOW@GMAIL.COM

**Gergely Neu**

*Universitat Pompeu Fabra, Barcelona, Spain*

GERGELY.NEU@GMAIL.COM

**Editors:** Alina Beygelzimer and Daniel Hsu

## Abstract

We consider a partial-feedback variant of the well-studied online PCA problem where a learner attempts to predict a sequence of  $d$ -dimensional vectors in terms of a quadratic loss, while only having limited feedback about the environment’s choices. We focus on a natural notion of bandit feedback where the learner only observes the loss associated with its own prediction. Based on the classical observation that this decision-making problem can be lifted to the space of density matrices, we propose an algorithm that is shown to achieve a regret of  $\tilde{O}(d^{3/2}\sqrt{T})$  after  $T$  rounds in the worst case. We also prove data-dependent bounds that improve on the basic result when the loss matrices of the environment have bounded rank or the loss of the best action is bounded. One version of our algorithm runs in  $O(d)$  time per trial which massively improves over every previously known online PCA method. We complement these results by a lower bound of  $\Omega(d\sqrt{T})$ .

**Keywords:** online PCA, bandit PCA, online linear optimization, phase retrieval

## 1. Introduction

Consider the problem of *phase retrieval* where one is interested in reconstructing a unit-norm vector  $\mathbf{x} \in \mathbb{R}^d$  up to a sign based on a number of noisy measurements of the form  $|\mathbf{w}_t^\top \mathbf{x}|^2$ . Such problems arise abundantly in numerous areas of science and engineering such as in X-ray crystallography, astronomy, and diffractive imaging (Millane, 1990). In the classical setting of phase retrieval, the measurement vectors  $\mathbf{w}_t$  are typically drawn i.i.d. from a distribution chosen before any measurements are taken (Fienup, 1982; Candès et al., 2013; Shechtman et al., 2015). In the present paper, we study a sequential decision-making framework generalizing this classical problem to situations where the measurements can be chosen adaptively and the sequence of hidden vectors can be chosen by an adversary.

Our formulation can be most accurately described as a partial-information variant of the well-studied problem of online principal component analysis (online PCA) (Warmuth and Kuzmin, 2006, 2008; Nie et al., 2016). In the basic version of the online PCA problem, the learner receives a sequence of input vectors  $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_T$ , and is tasked with projecting these vectors one by one to a sequence of one-dimensional hyperplanes represented by the rank-one projection matrices  $\mathbf{P}_t = \mathbf{w}_t \mathbf{w}_t^\top$  (with  $\|\mathbf{w}_t\| = 1$ ), in order to maximize the total squared norm of the projected inputs,  $\sum_t \|\mathbf{P}_t \mathbf{x}_t\|^2$ . Crucially, the learner selects each projection before observing the input vector, but nevertheless the input vector is fully revealed to the learner at the end of each round. In our problem setup, we remove this last assumption and assume that the learner *only observes the projection*

“gain”  $\|\mathbf{P}_t \mathbf{x}_t\|^2$ , but not the input vector  $\mathbf{x}_t$ . By analogy to the multi-armed bandit problem, we will refer to this setting as *bandit PCA*.

As already noted by Warmuth and Kuzmin (2006), the seemingly quadratic objective is in fact a linear function of the projection,  $\|\mathbf{P}_t \mathbf{x}_t\|^2 = \text{tr}(\mathbf{P}_t \mathbf{x}_t \mathbf{x}_t^\top)$ . Therefore, the bandit PCA problem can be reduced to a *linear bandit problem*, in which the learner plays with a rank-one projection matrix  $\mathbf{P}_t$ , the environment chooses a symmetric *loss matrix*  $\mathbf{L}_t = -\mathbf{x}_t \mathbf{x}_t^\top$ , and the learner suffers and observes loss  $\text{tr}(\mathbf{P}_t \mathbf{L}_t)$ . Using a generic algorithm for linear bandits, the continuous version of the Exponential Weights algorithm (Dani et al., 2008; Bubeck and Eldan, 2015; van der Hoeven et al., 2018), one can achieve a regret bound of order  $\mathcal{O}(p\sqrt{T \ln T})$ , where  $p$  is the dimension of the action and loss spaces. Unfortunately, the algorithm is computationally inefficient as it needs to maintain and update a distribution over the continuous set of rank-one projection matrix. Furthermore, observe that  $p = \mathcal{O}(d^2)$  in our setup, so the regret bound is in fact quadratic in the dimension of the problem.

In this paper, we address both of the above shortcomings and propose an efficient algorithm for a generalization of the bandit PCA problem in which the adversary is allowed to play symmetric loss matrices of arbitrary rank. Our algorithm achieves a regret bound of  $\mathcal{O}(d\sqrt{rT \ln T})$ , where  $r$  is the average squared Frobenius norm of the loss matrices played by the environment (which is upper bounded by their maximal rank of these matrices). Our regret bound improves the one mentioned above by at least a factor of  $\sqrt{d}$ , and can achieve a factor of  $d$  improvement when the Frobenius norm of the losses is bounded by a constant (e.g., in the original PCA case when all  $\mathbf{L}_t$  have rank one). We complement our results with a lower bound of  $\Omega(d\sqrt{T})$ , leaving a factor of  $\sqrt{d}$  gap between the two bounds in general. An interesting consequence of our lower bound is that it formally confirms the intuition that the bandit PCA problem is *strictly harder* than the  $d$ -armed bandit problem where the minimax regret is of order  $\Theta(\sqrt{dT})$  (Auer et al., 2002; Audibert and Bubeck, 2010). These results are to be contrasted with the fact that the full-information online PCA problem is *exactly as hard* as the problem of prediction with expert advice, the minimax regret being of  $\Theta(\sqrt{T \log d})$  in both cases (Nie et al., 2016).

On the front of computational complexity, one version of our algorithm achieves a surprisingly massive improvement over every previously known online PCA algorithm. Specifically, our algorithm only requires  $\tilde{\mathcal{O}}(d)$  computation per iteration, amounting to *sublinear* runtime in the dimension of the action space  $p = \mathcal{O}(d^2)$ . This striking runtime complexity should be contrasted with the full-information setup, in which the regret-optimal algorithms can only guarantee  $\mathcal{O}(d^\omega)$ <sup>1</sup> per-round complexity for full-rank loss matrices (Warmuth and Kuzmin, 2008; Allen-Zhu and Li, 2017). In fact, full information algorithms all face the computational bottleneck of having to read out the entries of  $\mathbf{L}_t$ , which already takes  $\mathcal{O}(d^2)$  time (unless  $\mathbf{L}_t$  is a low-rank matrix). In contrast, our partial-information setup stipulates that nature computes and communicates the realized loss for the learner at no computational cost. We note that our algorithms can be readily adjusted to cope with noisy observations, which enables the use of fast randomized linear algebra methods for computing the losses.

Our algorithm is based on the generic algorithmic template of online mirror descent (OMD) (Nemirovski and Yudin, 1983; Beck and Teboulle, 2003; Hazan, 2015; Joulani et al., 2017). Similarly to the methods for the full-information variant of online PCA (Nie et al., 2016), the algorithm maintains in each trial  $t = 1, \dots, T$  a *density matrix*  $\mathbf{W}_t$  as a parameter, which is a positive

1. Time needed for matrix multiplication, which is also the complexity of eigendecomposition with distinct eigenvalues (Allen-Zhu and Li, 2017).

definite matrix with unit trace, and represents a mixture over rank-one projections. In each trial  $t$ , a projection  $\mathbf{w}_t \mathbf{w}_t^\top$  is sampled in such a way that its expectation matches the density matrix,  $\mathbb{E}[\mathbf{w}_t \mathbf{w}_t^\top] = \mathbf{W}_t$ . Based on the observed loss, the algorithm constructs an unbiased estimate  $\tilde{\mathbf{L}}_t$  of the unknown loss matrix  $\mathbf{L}_t$ , which is then used to update the density matrix to  $\mathbf{W}_{t+1}$ .

The recipe described above is standard in the bandit literature, with a few degrees of freedom in choosing the regularization function for OMD, the scheme for sampling  $\mathbf{w}_t$ , and the structure of the loss estimator  $\tilde{\mathbf{L}}_t$ . While it may appear tempting to draw inspiration from existing full-information online PCA algorithms to make these design choices, it turns out that none of the previously used techniques are applicable in our setting. In particular, the previously employed methods of sampling from a density matrix (Warmuth and Kuzmin, 2006, 2008) by selecting eigendirections with probabilities equal to the eigenvalues turns out to be insufficient, as it is only able to sense the diagonal elements of the loss matrix (when expressed in the eigensystem of the learner’s density matrix), making it impossible to construct an unbiased loss estimator. Therefore, our first key algorithmic tool is designing a more sophisticated sampling scheme for  $\mathbf{w}_t$  and a corresponding loss estimator. Furthermore, we observe that the standard choice of the quantum negative entropy as the OMD regularizer (Tsuda et al., 2005) fails to provide the desired regret bound, no matter what unbiased loss estimator is used. Instead, our algorithm is crucially based on using the negative log-determinant  $-\log \det(\mathbf{W})$  as the regularization function.

### 1.1. Related work

Our work is a direct extension of the line of research on online PCA initiated by Warmuth and Kuzmin (2006) and further studied by Warmuth and Kuzmin (2008); Nie et al. (2016). Online PCA is an instance of the more general class of online matrix prediction problems, where the goal of the learner is to minimize its regret against the best matrix prediction chosen in hindsight (Tsuda et al., 2005; Garber et al., 2015; Allen-Zhu and Li, 2017). Boutsidis et al. (2015) studied another flavor of the online PCA problem where the goal of the learner is to encode a sequence of high-dimensional input vectors in a smaller representation.

Besides the above-mentioned works on online matrix prediction with full information, there is little existing work on the problem under partial information. One notable exception is the work of Gonen et al. (2016) that considers a problem of reconstructing the top principal components of a sequence of vectors  $\mathbf{x}_t$  while observing  $r \geq 2$  arbitrarily chosen entries of the  $d$ -dimensional inputs. Gonen et al. propose an algorithm based on the Matrix Exponentiated Gradient method and analyze its sample complexity through regret analysis and an online-to-batch conversion. Their analysis is greatly facilitated by the observation model that effectively allows a decoupling of exploration and exploitation, since the loss of the algorithm is only very loosely related to the chosen observations. In contrast, our setting presents the learner with a much more challenging dilemma since the observations are strictly tied to the incurred losses, and our feedback only consists of a single real number instead of  $r \geq 2$ . This latter difference, while seemingly minor, can often result in a large gap between the attainable regret guarantees (Agarwal et al., 2010; Hu et al., 2016).

Another closely related problem setting dubbed “rank-1 bandits” was considered by Katariya et al. (2017), Kveton et al. (2017), and Jun et al. (2019). In these problems, the learner is tasked with choosing two  $d$ -dimensional decision vectors  $\mathbf{x}_t$  and  $\mathbf{y}_t$ , and obtains a reward that is a bilinear function of the chosen vectors:  $\mathbf{x}_t^\top \mathbf{R}_t \mathbf{y}_t$  for some matrix  $\mathbf{R}_t$ . The setup most closely related to ours is the one considered by Jun et al. (2019), who assume arbitrary action sets for the learner and

prove regret bounds of order  $d^{3/2}\sqrt{rT}$ , where  $r$  is the rank of the reward matrix. Notably, these results assume that  $\mathbf{R}_t$  is generated i.i.d. from some unknown distribution. These results are to be contrasted with our bounds of order  $d\sqrt{rT}$  that are proven for adversarially chosen loss matrices. Note however that the two results are not directly comparable due to the mismatch between the considered decision sets: our decision set is in some sense smaller but more complex due to the semidefinite constraint, whereas theirs is larger but has simpler constraints.

Our analysis heavily draws on the literature on non-stochastic multi-armed bandits [Auer et al. \(2002\)](#); [Audibert and Bubeck \(2010\)](#); [Bubeck and Cesa-Bianchi \(2012\)](#), and makes particular use of the regularization function commonly known as the *log-barrier* (or the *Burg entropy*) that has been recently applied with great success to solve a number of challenging bandit problems ([Foster et al., 2016](#); [Agarwal et al., 2017](#); [Bubeck et al., 2018](#); [Wei and Luo, 2018](#); [Luo et al., 2018](#)). Indeed, our log-determinant regularizer is a direct generalization of the log-barrier function to the case of matrix-valued predictions, where the induced Bregman divergence is often called Stein’s loss. This loss function is commonly used in covariance matrix estimation in statistics ([James and Stein, 1961](#)), and has been applied in a range of machine-learning problems such as online metric learning ([Davis et al., 2007](#); [Jain et al., 2009](#); [Kulis and Bartlett, 2010](#)) and online local learning ([Christiano, 2014](#); [Awasthi et al., 2015](#)).

Finally, let us comment on the close relationship between our setting and that of phase retrieval, already alluded to at the very beginning of this paper. Indeed, the connection is readily apparent by noticing that the quadratic gain  $|\mathbf{w}_t^\top \mathbf{x}|^2$  is equivalent to the projection gain  $\|\mathbf{P}_t^\top \mathbf{x}\|^2$ , amounting to a bandit PCA problem instance with loss matrix  $-\mathbf{x}\mathbf{x}^\top + \xi_t \mathbf{I}$ , where the last term serves to model observation noise. A typical goal of a phase retrieval algorithm is to output a vector  $\hat{\mathbf{x}}$  that minimizes the distance  $\delta(\hat{\mathbf{x}}, \mathbf{x}) = \min \|\mathbf{x} \pm \hat{\mathbf{x}}\|$  to the hidden signal  $\mathbf{x}$ . It is easy to show that our regret bounds of minimax order  $\sqrt{T}$  translate to upper bounds on  $\delta(\hat{\mathbf{x}}, \mathbf{x})$  of order  $T^{-1/4}$  through a simple online-to-batch conversion, matching early results on phase retrieval by [Eldar and Mendelson \(2014\)](#). However, more recent results show that the true minimax rates for phase retrieval are actually of  $\Theta(T^{-1/2})$  ([Lecué and Mendelson, 2015](#); [Cai et al., 2016](#)). This highlights that in some sense the online version of this problem is much harder in that minimax rates for the regret do not seem to directly translate to minimax rates on the excess risk under i.i.d. assumptions.

**Notation.**  $\mathcal{S}$  is the set of  $d \times d$  symmetric positive semidefinite (SPSD) matrices and  $\mathcal{W} \subset \mathcal{S}$  is the set of density matrices  $\mathbf{W}$  satisfying  $\text{tr}(\mathbf{W}) = 1$ . We will use the notation  $\langle \mathbf{A}, \mathbf{B} \rangle = \text{tr}(\mathbf{A}^\top \mathbf{B})$  for any two  $d \times d$  matrices  $\mathbf{A}$  and  $\mathbf{B}$ , and define the Frobenius norm of any matrix  $\mathbf{A}$  as  $\|\mathbf{A}\|_F = \sqrt{\langle \mathbf{A}, \mathbf{A} \rangle}$ . We will consider randomized iterative algorithms that interact with a possibly random environment, giving rise to a filtration  $(\mathcal{F}_t)_{t \geq 1}$ . We will often use the shorthand  $\mathbb{E}_t[\cdot] = \mathbb{E}[\cdot | \mathcal{F}_t]$  to denote expectations conditional on the interaction history.

## 2. Preliminaries

We consider a sequential decision-making problem where a *learner* interacts with its *environment* by repeating the following steps in a sequence of rounds  $t = 1, 2, \dots, T$ :

1. learner picks a vector  $\mathbf{w}_t \in \mathbb{R}^d$  with unit norm, possibly in a randomized way,
2. environment picks a loss matrix  $\mathbf{L}_t$  with spectral norm bounded by 1,
3. learner incurs *and observes* loss  $\langle \mathbf{w}_t \mathbf{w}_t^\top, \mathbf{L}_t \rangle = \text{tr}(\mathbf{w}_t \mathbf{w}_t^\top \mathbf{L}_t)$ .

Note that the crucial difference from the traditional setup of online PCA is that the learner does not get to observe the full loss matrix  $\mathbf{L}_t$ . We will make the most minimal assumptions about the environment: the loss  $\mathbf{L}_t$  in round  $t$  is allowed to depend on the entire interaction history except the last decision  $\mathbf{w}_t$  of the learner. In other words, we will consider algorithms that work against *non-oblivious* or *adaptive adversaries*.

The performance of the learner is measured in terms of the *total expected regret* (or, simply, the *regret*), which is the difference between the cumulative loss of the algorithm and that of a fixed action optimal in expectation:

$$\text{regret}_T = \max_{\mathbf{u}: \|\mathbf{u}\|=1} \sum_{t=1}^T \mathbb{E} [\langle \mathbf{w}_t \mathbf{w}_t^\top - \mathbf{u} \mathbf{u}^\top, \mathbf{L}_t \rangle],$$

where the expectation is with respect to the internal randomization of the learner<sup>2</sup>.

### 3. Algorithms and main results

This section presents our general algorithmic template, based on the generic algorithmic framework of online mirror descent (Nemirovski and Yudin, 1983; Beck and Teboulle, 2003; Hazan, 2015; Joulani et al., 2017). Such algorithms are crucially based on a choice of a differentiable convex regularization function  $R : \mathcal{S} \rightarrow \mathbb{R}$  and the associated Bregman divergence  $D_R : \mathcal{S} \times \mathcal{S} \rightarrow \mathbb{R}_+$  induced by  $R$ :

$$D(\mathbf{W} \parallel \mathbf{W}') = R(\mathbf{W}) - R(\mathbf{W}') - \langle \nabla R(\mathbf{W}'), (\mathbf{W} - \mathbf{W}') \rangle.$$

Our version of online mirror descent proceeds by choosing the initial density matrix as  $\mathbf{W}_1 = \frac{1}{d} \mathbf{I} \in \mathcal{W}$ , and then iteratively computing the sequence of density matrices

$$\mathbf{W}_{t+1} = \operatorname{argmin}_{\mathbf{W} \in \mathcal{W}} \left\{ \eta \langle \mathbf{W}, \tilde{\mathbf{L}}_t \rangle + D_R(\mathbf{W} \parallel \mathbf{W}_t) \right\}.$$

Here,  $\tilde{\mathbf{L}}_t \in \mathcal{S}$  is an estimate of the loss matrix  $\mathbf{L}_t$  chosen by the environment in round  $t$ . Having computed  $\mathbf{W}_t$ , the algorithm randomly draws the unit-norm vector  $\mathbf{w}_t$  satisfying  $\mathbb{E}_t[\mathbf{w}_t \mathbf{w}_t^\top] = (1 - \gamma) \mathbf{W}_t + \frac{\gamma}{d} \mathbf{I}$ , where the latter term is added to prevent the eigenvalues of the covariance matrix from approaching 0. This effect is modulated by the parameter  $\gamma \in [0, 1]$  that we will call the *exploration rate*. The main challenges posed by our particular setting are:

- finding a way to sample a unit-length vector  $\mathbf{w}_t$  satisfying  $\mathbb{E}_t[\mathbf{w}_t \mathbf{w}_t^\top] = (1 - \gamma) \mathbf{W}_t + \frac{\gamma}{d} \mathbf{I}$ ,
- constructing a suitable (hopefully unbiased) loss estimator  $\tilde{\mathbf{L}}_t$  based on the observed loss  $\ell_t = \langle \mathbf{w}_t \mathbf{w}_t^\top, \mathbf{L}_t \rangle$  and the vector  $\mathbf{w}_t$ ,
- finding a regularization function  $R$  that is well-adapted to the previous design choices.

It turns out that addressing each of these challenges will require some unusual techniques. The most crucial element is the choice of regularization function that we choose as the negative log-determinant  $R(\mathbf{W}) = -\log \det(\mathbf{W})$ , with its derivative given as  $-\mathbf{W}^{-1}$  and the associated Bregman divergence being

$$D_R(\mathbf{W} \parallel \mathbf{U}) = \operatorname{tr}(\mathbf{U}^{-1} \mathbf{W}) - \log \det(\mathbf{U}^{-1} \mathbf{W}) - d,$$

2. This definition of regret is sometimes called *pseudo-regret* (Bubeck and Cesa-Bianchi, 2012).

which is sometimes called *Stein's loss* in the literature, and coincides with the relative entropy between the distributions  $\mathcal{N}(0, \mathbf{W})$  and  $\mathcal{N}(0, \mathbf{U})$ . In contrast to the multi-armed bandit setting, here the choice of the right regularizer turns out to be much more subtle, as the standard choices of the quantum negative entropy (Tsuda et al., 2005) or matrix Tsallis entropy (Allen-Zhu et al., 2015) fail to provide the desired regret bound (a discussion on these issues is included in Appendix C).

For sampling the vector  $\mathbf{w}_t$ , a peculiar challenge in our problem is having to design a process that will allow constructing an unbiased estimator of the loss matrix  $\mathbf{L}_t$ . To this end, we propose two different sampling strategies along with their corresponding loss-estimation schemes based on the eigendecomposition of the density matrices. The two strategies will be later shown to achieve two distinct flavors of data-dependent regret bounds. We present the details of these sampling schemes below in a simplified notation: given the eigenvalue decomposition  $\mathbf{W} = \sum_i \lambda_i \mathbf{u}_i \mathbf{u}_i^\top$  of density matrix  $\mathbf{W}$ , the procedures sample  $\mathbf{w}$  such that  $\mathbb{E}[\mathbf{w} \mathbf{w}^\top] = \mathbf{W}$ , and construct the loss estimate  $\tilde{\mathbf{L}}$  for which  $\mathbb{E}[\tilde{\mathbf{L}}] = \mathbf{L}$ . Recall that we use  $\mathbf{W} = (1 - \gamma)\mathbf{W}_t + \frac{\gamma}{d}\mathbf{I}$  in the algorithm.

---

**Algorithm 1:** Online Mirror Descent for Bandit PCA
 

---

**Parameters :** learning rate  $\eta > 0$ , exploration rate  $\gamma \in [0, 1]$

**Initialization:**  $\mathbf{W}_1 \leftarrow \frac{\mathbf{I}}{d}$

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

eigendecompose  $\mathbf{W}_t = \sum_{i=1}^d \mu_i \mathbf{u}_i \mathbf{u}_i^\top$

$\boldsymbol{\lambda} \leftarrow (1 - \gamma)\boldsymbol{\mu} + \gamma(\frac{1}{d}, \dots, \frac{1}{d})$

$\tilde{\mathbf{L}}_t \leftarrow \mathbf{sample}(\boldsymbol{\lambda}, \{\mathbf{u}_i\}_{i=1}^d)$

$\mathbf{W}_{t+1} \leftarrow (\mathbf{W}_t^{-1} + \eta \tilde{\mathbf{L}}_t + \beta \mathbf{I})^{-1}$  with  $\beta$  such that  $\text{tr}(\mathbf{W}_{t+1}) = 1$

---



---

**Algorithm 2:** Dense sampling
 

---

**def sample** ( $\boldsymbol{\lambda}, \{\mathbf{u}_i\}_{i=1}^d$ ):

$B \sim \text{Bernoulli}(\frac{1}{2})$

**if**  $B = 1$  **then**

draw  $I \sim \boldsymbol{\lambda}$  and set  $\mathbf{w}_t \leftarrow \mathbf{u}_I$

**else**

draw  $\mathbf{s} \in \{-1, +1\}^d$  i.i.d. uniformly

$\mathbf{w}_t \leftarrow \sum_i s_i \sqrt{\lambda_i} \mathbf{u}_i$

play  $\mathbf{w}_t$  and observe  $\ell_t = \langle \mathbf{w}_t \mathbf{w}_t^\top, \mathbf{L}_t \rangle$

**if**  $B = 1$  **then**

$\tilde{\mathbf{L}}_t \leftarrow 2\ell_t \mathbf{W}_t^{-1/2} \mathbf{w}_t \mathbf{w}_t^\top \mathbf{W}_t^{-1/2}$

**else**

$\tilde{\mathbf{L}}_t \leftarrow \ell_t (\mathbf{W}_t^{-1} \mathbf{w}_t \mathbf{w}_t^\top \mathbf{W}_t^{-1} - \mathbf{W}_t^{-1})$

**return**  $\tilde{\mathbf{L}}_t$

---



---

**Algorithm 3:** Sparse sampling
 

---

**def sample** ( $\boldsymbol{\lambda}, \{\mathbf{u}_i\}_{i=1}^d$ ):

draw  $I, J \sim \boldsymbol{\lambda}$

**if**  $I = J$  **then**

$\mathbf{w}_t \leftarrow \mathbf{u}_I$

**else**

draw  $s \in \{-1, 1\}$  uniformly

$\mathbf{w}_t \leftarrow \frac{1}{\sqrt{2}}(\mathbf{u}_I + s\mathbf{u}_J)$

play  $\mathbf{w}_t$  and observe  $\ell_t = \langle \mathbf{w}_t \mathbf{w}_t^\top, \mathbf{L}_t \rangle$

**if**  $I = J$  **then**

$\tilde{\mathbf{L}}_t \leftarrow (\ell_t / \lambda_I^2) \mathbf{u}_I \mathbf{u}_I^\top$

**else**

$\tilde{\mathbf{L}}_t \leftarrow s\ell / (2\lambda_I \lambda_J) (\mathbf{u}_I \mathbf{u}_J^\top + \mathbf{u}_J \mathbf{u}_I^\top)$

**return**  $\tilde{\mathbf{L}}_t$

---

### 3.1. Dense sampling

Our first sampling scheme is composed of two separate sampling procedures, designed to sense and estimate the on- and off-diagonal entries of the loss matrix  $\mathbf{L}$  (when expressed in the eigensystem of  $\mathbf{W}$ ), respectively. Precisely, the procedure will first draw a Bernoulli random variable  $B$  with  $P(B = 1) = \frac{1}{2}$ , and sample  $\mathbf{w}$  depending on the outcome as follows:

- If  $B = 1$ , sample  $\mathbf{w}$  as one of the eigenvectors  $\mathbf{u}_I$  such that  $\mathbb{P}[I = i] = \lambda_i$ . This clearly gives  $\mathbb{E}[\mathbf{w}\mathbf{w}^\top] = \sum_{i=1}^d \lambda_i \mathbf{u}_i \mathbf{u}_i^\top = \mathbf{W}$ .
- If  $B = 0$ , draw i.i.d. uniform random signs  $\mathbf{s} = (s_1, \dots, s_d) \in \{-1, +1\}^d$  and sample  $\mathbf{w}$  as

$$\mathbf{w} = \sum_{i=1}^d s_i \sqrt{\lambda_i} \mathbf{u}_i.$$

Note that  $\|\mathbf{w}\| = 1$  and we have

$$\mathbb{E}[\mathbf{w}\mathbf{w}^\top] = \mathbb{E}_{\mathbf{s}} \left[ \sum_{ij} s_i s_j \sqrt{\lambda_i \lambda_j} \mathbf{u}_i \mathbf{u}_j^\top \right] = \sum_{ij} \underbrace{\mathbb{E}_{\mathbf{s}}[s_i s_j]}_{\delta_{ij}} \sqrt{\lambda_i \lambda_j} \mathbf{u}_i \mathbf{u}_j^\top = \sum_i \lambda_i \mathbf{u}_i \mathbf{u}_i^\top = \mathbf{W}.$$

The first method is the standard sampling procedure in the full-information version of online PCA. In the bandit case, this method turns out to be insufficient, as it only let us observe  $\langle \mathbf{u}_i \mathbf{u}_i^\top, \mathbf{L} \rangle = \mathbf{u}_i^\top \mathbf{L} \mathbf{u}_i$ , that is, the on-diagonal elements of the loss matrix  $\mathbf{L}$  expressed in the eigensystem of  $\mathbf{W}$ . On the other hand, the second method does sense the off-diagonal elements  $\mathbf{u}_i^\top \mathbf{L} \mathbf{u}_j$ , but misses the on-diagonal ones. Thus, a combination of the two methods is sufficient for recovering the entire matrix. We will refer to this sampling method as *dense* since it observes a dense linear combination of the off-diagonal elements of the matrix  $\mathbf{L}$ . Having observed  $\ell = \langle \mathbf{w}\mathbf{w}^\top, \mathbf{L} \rangle$ , we construct our estimates in the two cases corresponding to the outcome of the random coin flip  $B$  as follows:

$$\tilde{\mathbf{L}} = \begin{cases} 2\ell \mathbf{W}^{-1/2} \mathbf{w}\mathbf{w}^\top \mathbf{W}^{-1/2} & \text{if } B = 1, \\ \ell (\mathbf{W}^{-1} \mathbf{w}\mathbf{w}^\top \mathbf{W}^{-1} - \mathbf{W}^{-1}) & \text{if } B = 0. \end{cases}$$

The following lemma (proved in Appendix A.1) shows that the above-defined estimate is unbiased.

**Lemma 1** *The estimate  $\tilde{\mathbf{L}}_t$  defined through the dense sampling method satisfies  $\mathbb{E}_t \tilde{\mathbf{L}}_t = \mathbf{L}_t$ .*

### 3.2. Sparse sampling

Our second method is based on sampling two eigenvectors of  $\mathbf{W}$  with indices  $I$  and  $J$  independently from the same distribution satisfying  $\mathbb{P}[J = i] = \mathbb{P}[I = i] = \lambda_i$ . Then, when  $I = J$ , it selects  $\mathbf{w} = \mathbf{u}_I$ , whereas for  $I \neq J$ , it draws a uniform random sign  $s \in \{-1, 1\}$  and sets  $\mathbf{w} = \frac{1}{\sqrt{2}}(\mathbf{u}_I + s\mathbf{u}_J)$ . We refer to this procedure as *sparse* since the observed loss is a sparse linear combination of diagonal and off-diagonal elements. We first verify that this method indeed satisfies  $\mathbb{E}[\mathbf{w}\mathbf{w}^\top] = \mathbf{W}$ :

$$\begin{aligned} \mathbb{E}[\mathbf{w}\mathbf{w}^\top] &= \underbrace{\sum_i \lambda_i^2 \mathbf{u}_i \mathbf{u}_i^\top}_{\text{when } I=J} + \underbrace{\sum_{i \neq j} \lambda_i \lambda_j \frac{1}{2} \mathbb{E}_{\mathbf{s}}[(\mathbf{u}_i + s\mathbf{u}_j)(\mathbf{u}_i + s\mathbf{u}_j)^\top]}_{\text{when } I \neq J} \\ &= \sum_i \lambda_i^2 \mathbf{u}_i \mathbf{u}_i^\top + \frac{1}{2} \sum_{i \neq j} \lambda_i \lambda_j (\mathbf{u}_i \mathbf{u}_i^\top + \mathbf{u}_j \mathbf{u}_j^\top) = \sum_{ij} \lambda_i \lambda_j \mathbf{u}_i \mathbf{u}_i^\top = \sum_i \lambda_i \mathbf{u}_i \mathbf{u}_i^\top = \mathbf{W}, \end{aligned}$$

where in the second equality we used the fact that  $s^2 = 1$  and  $\mathbb{E}_{\mathbf{s}}[s] = 0$ . The loss estimate is constructed as follows:

$$\tilde{\mathbf{L}} = \begin{cases} \frac{\ell}{\lambda_I^2} \mathbf{u}_I \mathbf{u}_I^\top & \text{when } I = J, \\ \frac{s\ell}{2\lambda_I \lambda_J} (\mathbf{u}_I \mathbf{u}_J^\top + \mathbf{u}_J \mathbf{u}_I^\top) & \text{when } I \neq J. \end{cases}$$

As the following lemma shows, this estimate is also unbiased. The proof is found in Appendix A.2.

**Lemma 2** *The estimate  $\tilde{\mathbf{L}}_t$  defined through the sparse sampling method satisfies  $\mathbb{E}_t \tilde{\mathbf{L}}_t = \mathbf{L}_t$ .*

### 3.3. Upper bounds on the regret

We can now state our main results regarding the performance of our algorithm with the two sampling schemes. Our first result is a data-dependent regret bound for the dense sampling method.

**Theorem 3** *Let  $\eta \leq \frac{1}{2d}$  and  $\gamma = 0$ . The regret of Algorithm 1 with dense sampling satisfies*

$$\text{regret}_T \leq \frac{d \log T}{\eta} + \eta(d^2 + 1) \sum_{t=1}^T \mathbb{E} [\ell_t^2] + 2.$$

We can immediately derive the following worst-case guarantee from the above result:

**Corollary 4** *Let  $\eta = \min \left\{ \sqrt{\frac{\log T}{dT}}, \frac{1}{2d} \right\}$  and  $\gamma = 0$ . Then, the regret of Algorithm 1 with dense sampling satisfies*

$$\text{regret}_T = \mathcal{O} \left( d^{3/2} \sqrt{T \log T} \right)$$

**Proof** The claim is trivial when  $\sqrt{(\log T)/(dT)} \geq 1/2d$ . Otherwise we use Theorem 3 together with  $\ell_t^2 \leq 1$  and plug in the choice of  $\eta$ .  $\blacksquare$

It turns out that the above bound can be significantly improved if we make some assumptions about the losses. Specifically, when the losses are assumed to be non-negative and there is a known upper bound on the cumulative loss of the best action:  $\bar{L}_T^* \geq \min_{\mathbf{u}: \|\mathbf{u}\|=1} \sum_t \text{tr}(\mathbf{u}\mathbf{u}^\top \mathbf{L}_t)$ , a properly tuned variant of our algorithm satisfies the following first-order regret bound (proof in Appendix A.6):

**Corollary 5** *Assume that  $L_t$  is positive semidefinite for all  $t$  and  $\bar{L}_T^*$  is defined as above. Then for  $\eta = \min \left\{ \sqrt{\frac{\log T}{d\bar{L}_T^*}}, \frac{1}{4d^2} \right\}$ , the regret of Algorithm 1 with dense sampling satisfies*

$$\text{regret}_T = \mathcal{O} \left( d^{3/2} \sqrt{\bar{L}_T^* \log T} + d^3 \log T \right)$$

Let us now turn to the version of our algorithm that uses the sparse sampling scheme.

**Theorem 6** *Let  $\eta \leq \frac{1}{2d}$  and  $\gamma = \eta d$ . The regret of Algorithm 1 with sparse sampling satisfies*

$$\text{regret}_T \leq \frac{d \log T}{\eta} + 2\eta d + 2 + 8\eta d \sum_{t=1}^T \mathbb{E} [\|\mathbf{L}_t\|_F^2]$$

**Corollary 7** *Let  $r \geq \frac{1}{T} \sum_{t=1}^T \mathbb{E} [\|\mathbf{L}_t\|_F^2]$  be known to the algorithm. Then, for  $\eta = \min \left\{ \sqrt{\frac{\log T}{rT}}, \frac{1}{2d} \right\}$  and  $\gamma = d\eta$ , the regret of the algorithm with sparse sampling satisfies*

$$\text{regret}_T = \mathcal{O} \left( d\sqrt{rT \log T} \right)$$

**Proof** The claim is trivial when  $\sqrt{(\log T)/(rT)} \geq 1/2d$ . Otherwise we use Theorem (6) and plug in the choice of  $\eta$ .  $\blacksquare$

Note that since the spectral norm of the losses is bounded by 1, we have  $\|\mathbf{L}_t\|_F^2 \leq \text{rank}(\mathbf{L}_t)$ . Thus, for the classical online PCA problem in which  $\mathbf{L}_t = -\mathbf{x}_t \mathbf{x}_t^\top$ , the bound becomes  $\mathcal{O}(d\sqrt{T \log T})$ .



### 3.4. Lower bound on the regret

We also prove the following lower bound on the regret of any algorithm:

**Theorem 8** *There exists a sequence of full-rank loss matrices such that the regret of any algorithm is lower bounded as*

$$\text{regret}_T \geq \frac{1}{16} d \sqrt{T / \log T}.$$

The proof can be found in Appendix A.7. Note that, since the lower bound involves full-rank losses, there is gap of order  $\sqrt{d}$  between the lower bound and the upper bounds achieved by our algorithms.

## 4. Analysis

This section presents the proofs of our main results. We decompose the proofs into two main parts: one considering the regret of online mirror descent with general loss estimators, and another part that is specific to the loss estimators we propose. Due to space limitations, the most technical parts of the proofs are delegated to Appendix A.

For the general mirror descent analysis, it will be useful to rewrite the update in the following form:

$$\begin{aligned} \text{(update step)} \quad \widetilde{\mathbf{W}}_{t+1} &= \underset{\mathbf{W}}{\text{argmin}} \left\{ D_R(\mathbf{W} \| \mathbf{W}_t) + \eta \text{tr}(\mathbf{W} \widetilde{\mathbf{L}}_t) \right\}, \\ \text{(projection step)} \quad \mathbf{W}_{t+1} &= \underset{\mathbf{W} \in \mathcal{W}}{\text{argmin}} D_R(\mathbf{W} \| \widetilde{\mathbf{W}}_{t+1}), \end{aligned} \tag{1}$$

where  $\mathcal{W}$  is the set of density matrices. The unprojected solution  $\widetilde{\mathbf{W}}_{t+1}$  can be shown to satisfy the equality  $\nabla R(\widetilde{\mathbf{W}}_{t+1}) = \nabla R(\mathbf{W}_t) - \eta \widetilde{\mathbf{L}}_t$ , which gives<sup>3</sup>

$$\widetilde{\mathbf{W}}_{t+1} = \left( \mathbf{W}_t^{-1} + \eta \widetilde{\mathbf{L}}_t \right)^{-1} = \mathbf{W}_t^{1/2} \left( \mathbf{I} + \eta \mathbf{W}_t^{1/2} \widetilde{\mathbf{L}}_t \mathbf{W}_t^{1/2} \right)^{-1} \mathbf{W}_t^{1/2}. \tag{2}$$

Our analysis will rely on the result below that follows from a direct application of well-known regret bound of online mirror descent, and a standard trick to relate the regret on the true and estimated losses, originally due to Auer et al. (2002).

**Lemma 9** *For any  $\eta > 0$  and  $\gamma \in [0, 1]$ , the regret of Algorithm 1 satisfies*

$$\text{regret}_T \leq \frac{d \log T}{\eta} + 2\gamma T + 2 + (1 - \gamma) \sum_{t=1}^T \mathbb{E} \left[ \langle \mathbf{W}_t - \widetilde{\mathbf{W}}_{t+1}, \widetilde{\mathbf{L}}_t \rangle \right].$$

The proof is rather standard and is included in Appendix A.3. The main challenge is bounding the last term in the above equation. To ease further calculations, we rewrite this term with the help of the matrix  $\mathbf{B}_t = \mathbf{W}_t^{1/2} \widetilde{\mathbf{L}}_t \mathbf{W}_t^{1/2}$ . From the definition of  $\widetilde{\mathbf{W}}_{t+1}$ , we have

$$\widetilde{\mathbf{W}}_{t+1} = \mathbf{W}_t^{1/2} (\mathbf{I} + \eta \mathbf{B}_t)^{-1} \mathbf{W}_t^{1/2} = \mathbf{W}_t - \eta \mathbf{W}_t^{1/2} \mathbf{B}_t (\mathbf{I} + \eta \mathbf{B}_t)^{-1} \mathbf{W}_t^{1/2},$$

3. While we do not show it explicitly here, it will be apparent from the proof of Lemma 10 that this update is well-defined since  $\mathbf{W}_t^{-1} + \eta \widetilde{\mathbf{L}}_t$  is invertible under our choice of parameters.

where the second equality uses the easily-checked identity  $(\mathbf{I} + \mathbf{A})^{-1} = \mathbf{I} - \mathbf{A}(\mathbf{I} + \mathbf{A})^{-1}$ . Therefore, the term in question can be written as

$$\begin{aligned} \langle \mathbf{W}_t - \widetilde{\mathbf{W}}_{t+1}, \widetilde{\mathbf{L}}_t \rangle &= \eta \operatorname{tr} \left( \mathbf{W}_t^{1/2} \mathbf{B}_t (\mathbf{I} + \eta \mathbf{B}_t)^{-1} \mathbf{W}_t^{1/2} \widetilde{\mathbf{L}}_t \right) = \eta \operatorname{tr} \left( \mathbf{B}_t (\mathbf{I} + \eta \mathbf{B}_t)^{-1} \mathbf{B}_t \right) \\ &= \sum_{i=1}^d \eta \frac{b_{t,i}^2}{1 + \eta b_{t,i}}, \end{aligned} \quad (3)$$

where  $\{b_{t,i}\}_{i=1}^d$  are the eigenvalues of  $\mathbf{B}_t$ . It remains to bound the term on the right-hand side for the dense and the sparse sampling method.

We first state a bound for the dense sampling method:

**Lemma 10** *Suppose that  $\eta \leq \frac{1}{2d}$  and  $\gamma = 0$ . Then, the dense sampling method guarantees*

$$\langle \mathbf{W}_t - \widetilde{\mathbf{W}}_{t+1}, \widetilde{\mathbf{L}}_t \rangle \leq \begin{cases} \frac{8}{3} \eta \ell_t^2 & \text{if } B = 1, \\ 2\eta d^2 \ell_t^2 & \text{if } B = 0. \end{cases}$$

In particular, the expectation is bounded as

$$\mathbb{E}_t \left[ \langle \mathbf{W}_t - \widetilde{\mathbf{W}}_{t+1}, \widetilde{\mathbf{L}}_t \rangle \right] \leq \eta (d^2 + 1) \ell_t^2.$$

The proof can be found in Appendix A.4. Combining this result with Lemma 9 concludes the proof of Theorem 3. Regarding the sparse sampling method, our key result is the following bound:

**Lemma 11** *Suppose that  $\eta \leq \frac{1}{2d}$  and  $\gamma = \eta d$ . Then, the sparse sampling method guarantees*

$$\mathbb{E}_t \left[ \langle \mathbf{W}_t - \widetilde{\mathbf{W}}_{t+1}, \widetilde{\mathbf{L}}_t \rangle \right] \leq 8\eta d \|\mathbf{L}_t\|_F^2$$

For a proof, see Appendix A.5. The statement of Theorem 6 immediately follows from combining the above result with Lemma 9.

#### 4.1. Computational cost

The total computational cost of the algorithm equipped with dense sampling is dominated by a rank one update of the eigendecomposition of the parameter matrix  $\mathbf{W}_t$  in each trial, which can take  $O(d^3)$  time in the worst case. Surprisingly, the computational cost of the sparse sampling version of the algorithm is only  $\tilde{O}(d)$ . This is because in each trial  $t$ , the loss estimate  $\widetilde{\mathbf{L}}_t$  is constructed from up to two eigenvectors of  $\mathbf{W}_t$  and thus only the corresponding part of the eigendecomposition needs to be updated. Furthermore, the projection operation only affects the eigenvalues and can be accomplished by solving a simple line search problem. The details of the efficient implementation are given in Appendix B. The claimed  $\tilde{O}(d)$  per-iteration cost of the algorithm is without taking into account the time needed to compute the value of the observed loss (as otherwise reading out the entries of  $\mathbf{L}_t$  would already take  $O(d^2)$  time). In other words, we assume that the algorithm plays with  $\mathbf{w}_t$  and the nature computes and communicates the realized loss  $\ell_t = \operatorname{tr}(\mathbf{w}_t \mathbf{w}_t^\top \mathbf{L}_t)$  for the learner at no computational cost. This assumption can actually be verified for several problems of practical interest (such as the classical applications of phase retrieval), and helps to separate computational issues related to learning and loss computation in other cases.

## 5. Discussion

We conclude by discussing some aspects of our results and possible directions for future work.

**Possible extensions.** While we work with real and symmetric matrices throughout the paper, it is relatively straightforward to extend our techniques to work with more general losses. One important extension is considering complex vector spaces, which naturally arise in applications like phase retrieval or quantum information. Fortunately, our algorithms easily generalize to complex Hermitian matrices, essentially by replacing every transposition with a Hermitian conjugate, noting that the eigenvalues of Hermitian matrices remain real. The analysis can be carried out with obvious modifications (Kale, 2007; Aaronson et al., 2018), giving the same guarantees on the regret. Another possible extension is to consider the *k-PCA problem*, in which the learner chooses a rank- $k$  projection matrix in each trial. It would also be interesting to extend our algorithms and their analysis the case of asymmetric loss matrices  $\mathbf{L}_t \in \mathbb{R}^{m \times n}$ , where the learner chooses two vectors  $\mathbf{x}_t \in \mathbb{R}^n$  and  $\mathbf{y}_t \in \mathbb{R}^m$ , and observes loss  $\text{tr}(\mathbf{L}_t \mathbf{x}_t \mathbf{y}_t^\top)$ , corresponding to the setup studied by Jun et al. (2019). We note here that extending the basic full-information online PCA formalism is possible through a clever embedding of such  $m \times n$  matrices into symmetric  $(m+n) \times (m+n)$  matrices, as shown by (Warmuth, 2007; Hazan et al., 2017). We leave it to future research to verify whether such a reduction would also work in the partial-feedback case.

**Comparison with continuous exponential weights.** As mentioned in the introduction, the bandit PCA problem can be directly formalized as an instance of bandit linear optimization, and one can prove regret bounds of  $\tilde{\mathcal{O}}(d^2 \sqrt{T})$  by an application of the generic continuous Exponential Weights analysis (Dani et al., 2008; Bubeck and Eldan, 2015; van der Hoeven et al., 2018). However, there are two major computational challenges that one needs to face when running this algorithm: sampling the density matrices  $\mathbf{W}_t$  and the decision vectors  $\mathbf{w}_t$ , and constructing unbiased estimates for the losses. Very recently, it has been shown by Pacchiano et al. (2019) that one can sample and update the exponential-weights distribution in  $\mathcal{O}(d^4)$  time for the decision set we consider in this paper, leaving us with the second problem. While in principle it is possible to use the generic loss estimator used in the above works (and originally proposed by McMahan and Blum, 2004; Awerbuch and Kleinberg, 2004), it is unclear if this estimator can actually be computed in polynomial time since it involves inverting a linear operator over density matrices. Indeed, it is not clear if the linear operator itself can be computed in polynomial time, let alone its inverse. In contrast, our algorithms achieve regret bounds of  $\tilde{\mathcal{O}}(d^{3/2} \sqrt{T})$  in the worst case, and run in  $\tilde{\mathcal{O}}(d)$  time when using sparse sampling for loss estimation.

**The gap between the upper and lower bounds.** One unsatisfying aspect of our results is the gap of order  $\sqrt{d}$  between the upper and lower bounds. Indeed, while Algorithm 1 with sparse sampling guarantees a regret bound of order  $d\sqrt{T}$  on rank-1 losses, seemingly matching the lower bounds for this case, this upper bound is in fact not comparable to the lower bound since the latter is proved for *full-rank* loss matrices. It is yet unclear which one of the bounds is tight, and we pose it as an exciting open problem to determine the minimax regret in this setup. We believe, however, that the upper bounds for our algorithms cannot be improved, and achieving minimax regret would require a radically different approach if it is our lower bound that captures the correct scaling with  $d$ .

**High-probability bounds.** All our regret bounds proved in the paper hold on expectation. It is natural to ask if it is possible to adjust our techniques to yield bounds that hold with high probability.

Unfortunately, our attempts to prove such bounds were unsuccessful due to a limitation common to all known techniques for proving high-probability bounds. Briefly put, all known approaches (Auer et al., 2002; Bartlett et al., 2008; Audibert and Bubeck, 2010; Beygelzimer et al., 2011; Neu, 2015) are based on adjusting the unbiased loss estimates so that the loss of every action  $\mathbf{v}$  is slightly underestimated by a margin of  $\beta \mathbb{E}_t[\langle \mathbf{v} \mathbf{v}^\top, \tilde{\mathbf{L}}_t^2 \rangle]$  for some small  $\beta$  of order  $T^{-1/2}$  (see, e.g., Abernethy and Rakhlin, 2009 for a general discussion). While it is straightforward to bias our own estimates in the same way, this eventually leads to extra terms of order  $\beta \mathbb{E}_t[\langle \mathbf{W}_t, \tilde{\mathbf{L}}_t^2 \rangle]$  in the bound, which are impossible to control by a small enough upper bound, as shown in Appendix C. Thus, proving high-probability bounds in our setting seems to require a fundamentally new approach, and we pose solving this challenge as another interesting problem for future research.

**Data-dependent bounds.** Besides a worst-case bound of order  $d^{3/2} \sqrt{T}$  on the regret, we also provide further guarantees that improve over the above when the loss matrices satisfy certain conditions. This raises the question if it is possible to achieve further improvements under other assumptions on the environment. A particularly interesting question is whether or not it is possible to improve our bounds for i.i.d. loss matrices generated by a *spiked covariance model* (Johnstone, 2001), corresponding to the most commonly studied setting in our primary motivating example of phase retrieval (Candès et al., 2013; Lecué and Mendelson, 2015). Obtaining faster rates for this setup would account for the discrepancy between the minimax bounds for phase retrieval and those obtained by an online-to-batch conversion from our newly proved bounds. We hope that the results provided in the present paper will initiate a new line of research on online phase retrieval that will eventually yield algorithms that take full advantage of adaptively chosen measurements and outperform traditional approaches for phase retrieval.

## Acknowledgments

W. Kotłowski was supported by the Polish National Science Centre under grant No. 2016/22/E/ST6/00299. G. Neu has been supported by “la Caixa” Banking Foundation through the Junior Leader Postdoctoral Fellowship Programme and a Google Faculty Research Award. Parts of this work were done while G. Neu was a visiting researcher at Google Brain in Zürich.

## References

- Scott Aaronson, Xinyi Chen, Elad Hazan, Satyen Kale, and Ashwin Nayak. Online learning of quantum states. In *Advances in Neural Information Processing Systems 31 (NIPS)*, pages 8976–8986. 2018.
- Jacob D. Abernethy and Alexander Rakhlin. Beating the adaptive bandit with high probability. In *Proceedings of the 22nd Annual Conference on Learning Theory (COLT)*, 2009.
- Jacob D. Abernethy, Chansoo Lee, and Ambuj Tewari. Fighting bandits with a new kind of smoothness. In *Advances in Neural Information Processing Systems 28 (NIPS)*, pages 2197–2205. 2015.
- Alekh Agarwal, Ofer Dekel, and Lin Xiao. Optimal algorithms for online convex optimization with multi-point bandit feedback. In *Proceedings of the 23rd Annual Conference on Learning Theory (COLT)*, pages 28–40, 2010.

- Alekh Agarwal, Haipeng Luo, Behnam Neyshabur, and Robert E Schapire. Corraling a band of bandit algorithms. In *Proceedings of the 30th Annual Conference on Learning Theory (COLT)*, pages 12–38, 2017.
- Zeyuan Allen-Zhu and Yuanzhi Li. Follow the compressed leader: Faster online learning of eigenvectors and faster MMWU. In *Proceedings of the 34th International Conference on Machine Learning (ICML)*, pages 116–125, 2017.
- Zeyuan Allen-Zhu, Zhenyu Liao, and Lorenzo Orecchia. Spectral sparsification and regret minimization beyond matrix multiplicative updates. In *Proceedings of the 47th Annual ACM on Symposium on Theory of Computing (STOC)*, pages 237–245, 2015. Newer version available at <http://arxiv.org/abs/1506.04838>.
- Zeyuan Allen-Zhu, Yuanzhi Li, Aarti Singh, and Yining Wang. Near-optimal design of experiments via regret minimization. In *Proceedings of the 34th International Conference on Machine Learning (ICML)*, pages 126–135, 2017.
- Sanjeev Arora, Elad Hazan, and Satyen Kale. Fast algorithms for approximate semidefinite programming using the multiplicative weights update method. In *IEEE Symposium on Foundations of Computer Science, FOCS '05*, pages 339–348. IEEE Computer Society, 2005.
- Jean-Yves Audibert and Sébastien Bubeck. Regret bounds and minimax policies under partial monitoring. *Journal of Machine Learning Research*, 11:2785–2836, 2010.
- Peter Auer, Nicolò Cesa-Bianchi, Yoav Freund, and Robert E. Schapire. The nonstochastic multi-armed bandit problem. *SIAM J. Comput.*, 32(1):48–77, 2002. ISSN 0097-5397.
- Pranjal Awasthi, Moses Charikar, Kevin A. Lai, and Andrej Risteski. Label optimal regret bounds for online local learning. In *Proceedings of The 28th Conference on Learning Theory (COLT)*, pages 150–166, 2015.
- Baruch Awerbuch and Robert D. Kleinberg. Adaptive routing with end-to-end feedback: distributed learning and geometric approaches. In *Proceedings of the 36th Annual ACM Symposium on Theory of Computing*, pages 45–53, 2004.
- Peter L. Bartlett, Varsha Dani, Thomas Hayes, Sham Kakade, Alexander Rakhlin, and Ambuj Tewari. High probability regret bounds for online optimization. In *Proceedings of the 21st Annual Conference on Learning Theory (COLT)*, pages 335–342, 2008.
- Amir Beck and Marc Teboulle. Mirror descent and nonlinear projected subgradient methods for convex optimization. *Operations Research Letters*, 31(3):167–175, 2003.
- Dennis S. Bernstein. *Matrix Mathematics*. Princeton University Press, 2009.
- Alina Beygelzimer, John Langford, Lihong Li, Lev Reyzin, and Robert E. Schapire. Contextual bandit algorithms with supervised learning guarantees. In *Proceedings of the 14th International Conference on Artificial Intelligence and Statistics (AISTATS 2011)*, pages 19–26, 2011.
- Christos Boutsidis, Dan Garber, Zohar Karnin, and Edo Liberty. Online principal component analysis. In *Proceedings of the 26th annual ACM-SIAM symposium on Discrete algorithms (SODA)*, pages 887–901, 2015.

- Sébastien Bubeck and Nicolò Cesa-Bianchi. Regret analysis of stochastic and nonstochastic multi-armed bandit problems. *Foundations and Trends in Machine Learning*, 5(1):1–122, 2012.
- Sébastien Bubeck and Ronen Eldan. The entropic barrier: a simple and optimal universal self-concordant barrier. In *Proceedings of the 28th Annual Conference on Learning Theory (COLT)*, pages 279–279, 2015.
- Sébastien Bubeck, Michael Cohen, and Yuanzhi Li. Sparsity, variance and curvature in multi-armed bandits. In *Proceedings of the 29th International Conference on Algorithmic Learning Theory (ALT)*, pages 111–127, 2018.
- T. Tony Cai, Xiaodong Li, and Zongming Ma. Optimal rates of convergence for noisy sparse phase retrieval via thresholded wirtinger flow. *The Annals of Statistics*, 44(5):2221–2251, 2016.
- Emmanuel J. Candès, Thomas Strohmer, and Vladislav Voroninski. Phaselift: Exact and stable signal recovery from magnitude measurements via convex programming. *Communications on Pure and Applied Mathematics*, 66(8):1241–1274, 2013.
- Nicolò Cesa-Bianchi and Gábor Lugosi. *Prediction, Learning, and Games*. Cambridge University Press, New York, NY, USA, 2006.
- Paul Christiano. Online local learning via semidefinite programming. In *Proceedings of the 46th Annual ACM Symposium on Theory of Computing (STOC)*, pages 468–474, 2014.
- Alon Cohen, Tamir Hazan, and Tomer Koren. Tight bounds for bandit combinatorial optimization. In *Proceedings of the 30th Annual Conference on Learning Theory (COLT)*, pages 629–642, 2017.
- Varsha Dani, Thomas Hayes, and Sham Kakade. The price of bandit information for online optimization. In *Advances in Neural Information Processing Systems 20 (NIPS)*, pages 345–352, 2008.
- Jason V. Davis, Brian Kulis, Prateek Jain, Suvrit Sra, and Inderjit S. Dhillon. Information-theoretic metric learning. In *Proceedings of the 24th international conference on Machine learning*, pages 209–216. ACM, 2007.
- Luc Devroye. *Non-Uniform Random Variate Generation*. Springer-Verlag, 1986.
- Yonina C. Eldar and Shahar Mendelson. Phase retrieval: Stability and recovery guarantees. *Applied and Computational Harmonic Analysis*, 36(3):473–494, 2014.
- James R. Fienup. Phase retrieval algorithms: a comparison. *Applied optics*, 21(15):2758–2769, 1982.
- Dylan J. Foster, Zhiyuan Li, Thodoris Lykouris, Karthik Sridharan, and Eva Tardos. Learning in games: Robustness of fast convergence. In *Advances in Neural Information Processing Systems 29 (NIPS)*, pages 4734–4742, 2016.
- Dan Garber, Elad Hazan, and Tengyu Ma. Online learning of eigenvectors. In *Proceedings of the 32nd International Conference on Machine Learning (ICML)*, pages 560–568, 2015.

- Alon Gonen, Dan Rosenbaum, Yonina C Eldar, and Shai Shalev-Shwartz. Subspace learning with partial information. *The Journal of Machine Learning Research*, 17(1):1821–1841, 2016.
- Ming Gu and Stanley C. Eisenstat. A stable and efficient algorithm for the rank-one modification of the symmetric eigenproblem. *SIAM J. Matrix Anal. Appl.*, 15(4):1266–1276, 1994.
- Elad Hazan. Introduction to online convex optimization. *Foundations and Trends in Optimization*, 2(3–4):157–325, 2015.
- Elad Hazan, Satyen Kale, and Shai Shalev-Shwartz. Near-optimal algorithms for online matrix prediction. *SIAM J. Comput.*, 46(2):744–773, 2017.
- Xiaowei Hu, LA Prashanth, András György, and Csaba Szepesvári. (Bandit) convex optimization with biased noisy gradient oracles. In *Artificial Intelligence and Statistics*, pages 819–828, 2016.
- Prateek Jain, Brian Kulis, Inderjit S. Dhillon, and Kristen Grauman. Online metric learning and fast similarity search. In *Advances in neural information processing systems 21 (NIPS)*, pages 761–768, 2009.
- William James and Charles Stein. Estimation with quadratic loss. In *Proceedings of the Fourth Berkeley Symposium on Mathematical Statistics and Probability, Volume 1: Contributions to the Theory of Statistics*, pages 361–379. University of California Press, 1961.
- Iain M. Johnstone. On the distribution of the largest eigenvalue in principal components analysis. *Annals of statistics*, pages 295–327, 2001.
- Pooria Joulani, András György, and Csaba Szepesvári. A modular analysis of adaptive (non-) convex optimization: Optimism, composite objectives, and variational bounds. In *Proceedings of the 28th International Conference on Algorithmic Learning Theory (ALT)*, pages 681–720, 2017.
- Kwang-Sung Jun, Rebecca Willett, Stephen Wright, and Robert Nowak. Bilinear bandits with low-rank structure. In *Proceedings of the 36th International Conference on Machine Learning (ICML)*, pages 3163–3172, 2019.
- Satyen Kale. *Efficient Algorithms Using The Multiplicative Weights Update Method*. PhD thesis, Princeton University, 2007.
- Sumeet Katariya, Branislav Kveton, Csaba Szepesvári, Claire Vernade, and Zheng Wen. Stochastic rank-1 bandits. In *Artificial Intelligence and Statistics*, pages 392–401, 2017.
- Brian Kulis and Peter L. Bartlett. Implicit online learning. In *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, pages 575–582, 2010.
- Branislav Kveton, Csaba Szepesvári, Anup Rao, Zheng Wen, Yasin Abbasi-Yadkori, and S. Muthukrishnan. Stochastic low-rank bandits. *arXiv preprint arXiv:1712.04644*, 2017.
- Tor Lattimore and Csaba Szepesvári. Bandit algorithms. <http://downloads.tor-lattimore.com/banditbook/book.pdf>, 2019.

- Guillaume Lecué and Shahar Mendelson. Minimax rate of convergence and the performance of empirical risk minimization in phase retrieval. *Electronic Journal of Probability*, 20(57):1–29, 2015.
- Adrian S. Lewis and Hristo S. Sendov. Quadratic expansions of spectral functions. *Linear Algebra and its Applications*, 340:97–121, 2002.
- Haipeng Luo, Chen-Yu Wei, and Kai Zheng. Efficient online portfolio with logarithmic regret. In *Advances in Neural Information Processing Systems 31 (NIPS)*, pages 8245–8255, 2018.
- H. Brendan McMahan and Avrim Blum. Online geometric optimization in the bandit setting against an adaptive adversary. In *Proceedings of the 17th Annual Conference on Learning Theory (COLT)*, pages 109–123, 2004.
- Rick P. Millane. Phase retrieval in crystallography and optics. *JOSA A*, 7(3):394–411, 1990.
- Arkadi Nemirovski and David Yudin. *Problem Complexity and Method Efficiency in Optimization*. Wiley Interscience, 1983.
- Gergely Neu. Explore no more: Improved high-probability regret bounds for non-stochastic bandits. In *Advances in Neural Information Processing Systems 28 (NIPS)*, pages 3150–3158, 2015.
- Jiazhong Nie, Wojciech Kotłowski, and Manfred K. Warmuth. Online PCA with optimal regret. *Journal of Machine Learning Research*, 17(173):1–49, 2016.
- Aldo Pacchiano, Niladri Chatterji, and Peter Bartlett. Online learning with kernel losses. In *Proceedings of the 36th International Conference on Machine Learning (ICML)*, pages 971–980, 2019.
- Alexander Rakhlin. Lecture notes on online learning. Available at: [http://www.mit.edu/~rakhlin/papers/online\\_learning.pdf](http://www.mit.edu/~rakhlin/papers/online_learning.pdf), 2008.
- Shai Shalev-Shwartz. Online learning and online convex optimization. *Foundations and Trends in Machine Learning*, 4(2):107–194, 2011.
- Ohad Shamir. On the complexity of bandit linear optimization. In *Proceedings of The 28th Conference on Learning Theory (COLT)*, pages 1523–1551, 2015.
- Yoav Shechtman, Yonina C. Eldar, Oren Cohen, Henry Nicholas Chapman, Jianwei Miao, and Mordechai Segev. Phase retrieval with application to optical imaging: a contemporary overview. *IEEE signal processing magazine*, 32(3):87–109, 2015.
- Koji Tsuda, Gunnar Rätsch, and Manfred K. Warmuth. Matrix exponentiated gradient updates for on-line learning and Bregman projections. *Journal of Machine Learning Research*, 6:995–1018, 2005.
- Dirk van der Hoeven, Tim van Erven, and Wojciech Kotłowski. The many faces of exponential weights in online learning. In *Proceedings of the 31st Conference On Learning Theory*, volume 75 of *Proceedings of Machine Learning Research*, pages 2067–2092. PMLR, 2018.



Manfred K. Warmuth. Winnowing subspaces. In *Proceedings of the 24th International Conference on Machine Learning (ICML)*, pages 999–1006, 2007.

Manfred K. Warmuth and Dima Kuzmin. Online variance minimization. In *Proceedings of the 19th Annual Conference on Learning Theory (COLT)*, pages 514–528, 2006.

Manfred K. Warmuth and Dima Kuzmin. Randomized online PCA algorithms with regret bounds that are logarithmic in the dimension. *Journal of Machine Learning Research*, 9:2287–2320, 2008.

Chen-Yu Wei and Haipeng Luo. More adaptive algorithms for adversarial bandits. In *Proceedings of the 31st Conference On Learning Theory (COLT)*, pages 1263–1291, 2018.

Andrew Chi-Chin Yao. Probabilistic computations: Toward a unified measure of complexity. In *18th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 222–227, 1977.

## Appendix A. Ommitted proofs

### A.1. The proof of Lemma 1

For the proof, let us define  $L_{ij} = \mathbf{u}_i^\top \mathbf{L} \mathbf{u}_j$  and note that  $\mathbf{L} = \sum_{i,j} L_{ij} \mathbf{u}_i \mathbf{u}_j^\top$ . In the case when  $B = 1$ , we have

$$\mathbb{E}[\ell \mathbf{w} \mathbf{w}^\top | B = 1] = \mathbb{E}[\text{tr}(\mathbf{w} \mathbf{w}^\top \mathbf{L}) \mathbf{w} \mathbf{w}^\top | B = 1] = \sum_{i=1}^d \lambda_i \text{tr}(\mathbf{u}_i \mathbf{u}_i^\top \mathbf{L}) \mathbf{u}_i \mathbf{u}_i^\top = \sum_{i=1}^d \lambda_i L_{ii} \mathbf{u}_i \mathbf{u}_i^\top,$$

and thus

$$\mathbb{E}[\tilde{\mathbf{L}} | B = 1] = 2 \mathbf{W}^{-1/2} \left( \sum_{i=1}^d \lambda_i L_{ii} \mathbf{u}_i \mathbf{u}_i^\top \right) \mathbf{W}^{-1/2} = 2 \sum_{i=1}^d L_{ii} \mathbf{u}_i \mathbf{u}_i^\top,$$

where we used the fact that  $\mathbf{u}_i$  is the eigenvector of  $\mathbf{W}$ , so  $\mathbf{W}^{-1/2} \mathbf{u}_i = \lambda_i^{-1/2} \mathbf{u}_i$ .

When  $B = 0$ , we have:

$$\begin{aligned} \mathbb{E}[\text{tr}(\mathbf{w} \mathbf{w}^\top \mathbf{L}) \mathbf{w} \mathbf{w}^\top | B = 0] &= \mathbb{E}_{\mathbf{s}} \left[ \text{tr} \left( \sum_{ij} s_i s_j \sqrt{\lambda_i \lambda_j} \mathbf{u}_i \mathbf{u}_j^\top \mathbf{L} \right) \sum_{km} s_k s_m \sqrt{\lambda_k \lambda_m} \mathbf{u}_k \mathbf{u}_m^\top \right] \\ &= \mathbb{E}_{\mathbf{s}} \left[ \left( \sum_{ij} s_i s_j \sqrt{\lambda_i \lambda_j} L_{ij} \right) \left( \sum_{km} s_k s_m \sqrt{\lambda_k \lambda_m} \mathbf{u}_k \mathbf{u}_m^\top \right) \right] \\ &= \sum_{ijkm} \mathbb{E}_{\mathbf{s}} [s_i s_j s_k s_m] \sqrt{\lambda_i \lambda_j \lambda_k \lambda_m} L_{ij} \mathbf{u}_k \mathbf{u}_m^\top. \end{aligned}$$

Now,  $\mathbb{E}_{\mathbf{s}} [s_i s_j s_k s_m]$  is zero if one of the indices is a non-duplicate, such as the case  $i \notin \{j, k, m\}$ . The four cases where  $\mathbb{E}_{\mathbf{s}} [s_i s_j s_k s_m] = 1$  are the following: (I)  $i = j = k = m$ , (II)  $i = j$ ,  $k = m \neq i$ , (III)  $i = k$ ,  $j = m \neq i$ , (IV)  $i = m$ ,  $k = j \neq i$ . Considering these cases separately, we get

$$\begin{aligned} \mathbb{E}[\text{tr}(\mathbf{w} \mathbf{w}^\top \mathbf{L}) \mathbf{w} \mathbf{w}^\top | B = 0] &= \underbrace{\sum_{ij} \lambda_i \lambda_j L_{ii} \mathbf{u}_j \mathbf{u}_j^\top}_{\text{(I) + (II)}} + \underbrace{\sum_{i \neq j} \lambda_i \lambda_j L_{ij} \mathbf{u}_i \mathbf{u}_j^\top}_{\text{(III)}} + \underbrace{\sum_{i \neq j} \lambda_i \lambda_j L_{ij} \mathbf{u}_j \mathbf{u}_i^\top}_{\text{(IV)}} \\ &= \mathbf{W} \sum_i \lambda_i L_{ii} + 2 \sum_{ij} \lambda_i \lambda_j L_{ij} \mathbf{u}_i \mathbf{u}_j^\top - 2 \sum_i \lambda_i^2 L_{ii} \mathbf{u}_i \mathbf{u}_i^\top. \end{aligned}$$

Multiplying the above with  $\mathbf{W}^{-1}$  from both sides gives

$$\mathbf{W}^{-1} \mathbb{E}[\text{tr}(\mathbf{w} \mathbf{w}^\top \mathbf{L}) \mathbf{w} \mathbf{w}^\top | B = 0] \mathbf{W}^{-1} = \underbrace{\mathbf{W}^{-1} \sum_i \lambda_i L_{ii}}_{\text{tr}(\mathbf{W} \mathbf{L})} + 2 \underbrace{\sum_{ij} L_{ij} \mathbf{u}_i \mathbf{u}_j^\top}_{=\mathbf{L}} - 2 \sum_i L_{ii} \mathbf{u}_i \mathbf{u}_i^\top.$$

Furthermore, we clearly have  $\mathbb{E}[\ell | B = 0] = \text{tr}(\mathbb{E}[\mathbf{w} \mathbf{w}^\top | B = 0] \mathbf{L}) = \text{tr}(\mathbf{W} \mathbf{L})$ . Therefore using the definition of  $\tilde{\mathbf{L}}$ , we get

$$\begin{aligned} \mathbb{E}[\tilde{\mathbf{L}} | B = 0] &= \mathbf{W}^{-1} \text{tr}(\mathbf{W} \mathbf{L}) + 2 \mathbf{L} - 2 \sum_i L_{ii} \mathbf{u}_i \mathbf{u}_i^\top - \mathbf{W}^{-1} \mathbb{E}[\ell | B = 0] \\ &= 2 \left( \mathbf{L} - \sum_i L_{ii} \mathbf{u}_i \mathbf{u}_i^\top \right). \end{aligned}$$

Putting the two cases concludes the proof as

$$\mathbb{E} [\tilde{\mathbf{L}}] = \frac{1}{2} \mathbb{E} [\tilde{\mathbf{L}} | B = 1] + \frac{1}{2} \mathbb{E} [\tilde{\mathbf{L}} | B = 0] = \sum_{i=1}^d L_{ii} \mathbf{u}_i \mathbf{u}_i^\top + \left( \mathbf{L} - \sum_i L_{ii} \mathbf{u}_i \mathbf{u}_i^\top \right) = \mathbf{L}.$$

■

## A.2. The proof of Lemma 2

We remind that the loss estimate is constructed as:

$$\tilde{\mathbf{L}} = \begin{cases} \frac{\ell}{\lambda_I^2} \mathbf{u}_I \mathbf{u}_I^\top & \text{when } I = J, \\ \frac{\ell s}{2\lambda_I \lambda_J} (\mathbf{u}_I \mathbf{u}_J^\top + \mathbf{u}_J \mathbf{u}_I^\top) & \text{when } I \neq J. \end{cases}$$

We check that the estimate of the loss is unbiased. Let  $L_{ij} = \mathbf{u}_i^\top \mathbf{L} \mathbf{u}_j$ . We have:

$$\begin{aligned} \mathbb{E} [\tilde{\mathbf{L}}] &= \underbrace{\sum_i \lambda_i^2 \operatorname{tr}(\mathbf{u}_i \mathbf{u}_i^\top \mathbf{L}) \frac{1}{\lambda_i^2} \mathbf{u}_i \mathbf{u}_i^\top}_{\text{when } I=J} \\ &\quad + \underbrace{\sum_{i \neq j} \lambda_i \lambda_j \mathbb{E}_s \left[ \operatorname{tr} \left( \frac{1}{2} (\mathbf{u}_i + s \mathbf{u}_j) (\mathbf{u}_i + s \mathbf{u}_j)^\top \mathbf{L} \right) \frac{s}{2\lambda_i \lambda_j} (\mathbf{u}_i \mathbf{u}_j^\top + \mathbf{u}_j \mathbf{u}_i^\top) \right]}_{\text{when } I \neq J} \\ &= \sum_i L_{ii} \mathbf{u}_i \mathbf{u}_i^\top + \frac{1}{4} \sum_{i \neq j} (L_{ii} + L_{jj}) \underbrace{\mathbb{E}_s [s]}_{=0} (\mathbf{u}_i \mathbf{u}_j^\top + \mathbf{u}_j \mathbf{u}_i^\top) + \frac{1}{2} \sum_{i \neq j} L_{ij} (\mathbf{u}_i \mathbf{u}_j^\top + \mathbf{u}_j \mathbf{u}_i^\top) \\ &= \sum_{ij} L_{ij} \mathbf{u}_i \mathbf{u}_j^\top = \mathbf{L}, \end{aligned}$$

where in the second inequality we used the fact that  $s^2 = 1$ .

■

## A.3. The proof of Lemma 9

We start with the well-known result regarding the regret of mirror descent (see, e.g., [Rakhlin, 2008](#)). We include the simple proof in for completeness.

**Lemma 12** *For any  $\mathbf{U} \in \mathcal{S}$ , the following inequality holds:*

$$\sum_{t=1}^T \langle \mathbf{W}_t - \mathbf{U}, \tilde{\mathbf{L}}_t \rangle \leq \frac{D_R(\mathbf{U} \| \mathbf{W}_1)}{\eta} + \sum_{t=1}^T \langle \mathbf{W}_t - \tilde{\mathbf{W}}_{t+1}, \tilde{\mathbf{L}}_t \rangle.$$

**Proof** We start from the following well-known identity<sup>4</sup> that holds for for any three SPSD matrices  $\mathbf{U}, \mathbf{V}, \mathbf{W}$ :

$$D_R(\mathbf{U} \| \mathbf{V}) + D_R(\mathbf{V} \| \mathbf{W}) = D_R(\mathbf{U} \| \mathbf{W}) + \langle \mathbf{U} - \mathbf{V}, \nabla R(\mathbf{W}) - \nabla R(\mathbf{V}) \rangle.$$

4. This easily proven result is sometimes called the “three-points identity”.

Taking  $\mathbf{W} = \mathbf{W}_t$  and  $\mathbf{V} = \widetilde{\mathbf{W}}_{t+1}$  and using that  $D_R(\mathbf{V} \parallel \mathbf{W}) \geq 0$  gives

$$D_R(\mathbf{U} \parallel \widetilde{\mathbf{W}}_{t+1}) \leq D_R(\mathbf{U} \parallel \mathbf{W}_t) + \eta \langle \mathbf{U} - \widetilde{\mathbf{W}}_{t+1}, \widetilde{\mathbf{L}}_t \rangle.$$

Since  $D(\mathbf{U} \parallel \widetilde{\mathbf{W}}_{t+1}) \geq D(\mathbf{U} \parallel \mathbf{W}_{t+1})$  by the Generalized Pythagorean Inequality, we get

$$\eta \langle \widetilde{\mathbf{W}}_{t+1} - \mathbf{U}, \widetilde{\mathbf{L}}_t \rangle \leq D_R(\mathbf{U} \parallel \mathbf{W}_t) - D_R(\mathbf{U} \parallel \mathbf{W}_{t+1}).$$

Reordering and adding  $\langle \mathbf{W}_t, \widetilde{\mathbf{L}}_t \rangle$  to both sides gives

$$\langle \mathbf{W}_t - \mathbf{U}, \widetilde{\mathbf{L}}_t \rangle \leq \langle \mathbf{W}_t - \widetilde{\mathbf{W}}_{t+1}, \widetilde{\mathbf{L}}_t \rangle + \frac{1}{\eta} D_R(\mathbf{U} \parallel \mathbf{W}_t) - \frac{1}{\eta} D_R(\mathbf{U} \parallel \mathbf{W}_{t+1}).$$

Summing up for all  $t$  and noticing that  $D_R(\mathbf{U} \parallel \mathbf{W}_{T+1}) \geq 0$  concludes the proof.  $\blacksquare$

**Proof** (of Lemma 9). We start with relating the quantity on the left-hand side of statement in Lemma 12 to the regret of the algorithm. To this end, observe that the unbiasedness of  $\widetilde{\mathbf{L}}_t$  and the conditional independence of  $\widetilde{\mathbf{L}}_t$  on  $\mathbf{W}_t$  ensures that

$$(1 - \gamma) \mathbb{E}_t [\langle \mathbf{W}_t, \widetilde{\mathbf{L}}_t \rangle] = \langle (1 - \gamma) \mathbf{W}_t, \mathbf{L}_t \rangle = \mathbb{E}_t [\langle \mathbf{w}_t \mathbf{w}_t, \mathbf{L}_t \rangle] - \frac{\gamma}{d} \langle \mathbf{I}, \mathbf{L}_t \rangle,$$

where we also used the fact that  $\mathbf{w}_t$  is sampled so that  $\mathbb{E}_t [\mathbf{w}_t \mathbf{w}_t] = (1 - \gamma) \mathbf{W}_t + \frac{\gamma}{d} \mathbf{I}$  is satisfied. Similarly, for any fixed  $\mathbf{U}$  it holds  $\mathbb{E}_t [\langle \mathbf{U}, \widetilde{\mathbf{L}}_t \rangle] = \langle \mathbf{U}, \mathbf{L}_t \rangle$ . Using these relation results in

$$\mathbb{E}_t [\langle \mathbf{w}_t \mathbf{w}_t - \mathbf{U}, \mathbf{L}_t \rangle] = (1 - \gamma) \mathbb{E}_t [\langle \mathbf{W}_t - \mathbf{U}, \widetilde{\mathbf{L}}_t \rangle] + \gamma \left\langle \frac{\mathbf{I}}{d} - \mathbf{U}, \mathbf{L}_t \right\rangle. \quad (4)$$

Since  $\mathbf{L}_t$  has spectral norm bounded by 1, the last term on the right-hand side can be bounded by:

$$\left\langle \frac{\mathbf{I}}{d} - \mathbf{U}, \mathbf{L}_t \right\rangle \leq \left\| \frac{\mathbf{I}}{d} - \mathbf{U} \right\|_1 \|\mathbf{L}_t\|_\infty \leq \text{tr} \left( \frac{\mathbf{I}}{d} \right) + \text{tr}(\mathbf{U}) = 2$$

Using the above bound in (4), summing over trials and taking marginal expectation on both sides gives:

$$\begin{aligned} \sum_{t=1}^T \mathbb{E} [\langle \mathbf{w}_t \mathbf{w}_t - \mathbf{U}, \mathbf{L}_t \rangle] &\leq (1 - \gamma) \sum_{t=1}^T \mathbb{E} [\langle \mathbf{W}_t - \mathbf{U}, \widetilde{\mathbf{L}}_t \rangle] + 2\gamma T \\ &\leq (1 - \gamma) \frac{D_R(\mathbf{U} \parallel \mathbf{W}_1)}{\eta} + (1 - \gamma) \sum_{t=1}^T \mathbb{E} [\langle \mathbf{W}_t - \widetilde{\mathbf{W}}_{t+1}, \widetilde{\mathbf{L}}_t \rangle] + 2\gamma T, \end{aligned} \quad (5)$$

where the second inequality is from Lemma 12. One minor challenge is that the first term on the right-hand side of (5) is infinite for a “pure” comparator  $\mathbf{u}\mathbf{u}^\top$ . To deal with this issues, for any  $\mathbf{U}$

define the smoothed comparator  $\tilde{\mathbf{U}} = (1 - \theta)\mathbf{U} + \frac{\theta}{d}\mathbf{I}$  for some  $\theta \in [0, 1]$ . Using  $\mathbf{W}_1 = \frac{1}{d}\mathbf{I}$ , we have:

$$\begin{aligned} D_R(\tilde{\mathbf{U}} \parallel \mathbf{W}_1) &= \log \det \left( \frac{\mathbf{I}}{d} \right) - \log \det \left( (1 - \theta)\mathbf{U} + \frac{\theta}{d}\mathbf{I} \right) + d \operatorname{tr}(\tilde{\mathbf{U}}) - d \\ &\leq \log \det \left( \frac{\mathbf{I}}{d} \right) - \log \det \left( \frac{\theta}{d}\mathbf{I} \right) = d \log(1/\theta). \end{aligned}$$

Using (5) with the smoothed comparator  $\tilde{\mathbf{U}}$  gives:

$$\sum_{t=1}^T \mathbb{E} \left[ \langle \mathbf{w}_t \mathbf{w}_t - \tilde{\mathbf{U}}, \mathbf{L}_t \rangle \right] \leq (1 - \gamma) \frac{d \log(1/\theta)}{\eta} + (1 - \gamma) \sum_{t=1}^T \mathbb{E} \left[ \langle \mathbf{W}_t - \tilde{\mathbf{W}}_{t+1}, \tilde{\mathbf{L}}_t \rangle \right] + 2\gamma T.$$

Now, since:

$$\langle \mathbf{w}_t \mathbf{w}_t - \tilde{\mathbf{U}}, \mathbf{L}_t \rangle = \langle \mathbf{w}_t \mathbf{w}_t - \mathbf{U}, \mathbf{L}_t \rangle + \theta \left\langle \frac{\mathbf{I}}{d} - \mathbf{U}, \mathbf{L}_t \right\rangle \geq \langle \mathbf{w}_t \mathbf{w}_t - \mathbf{U}, \mathbf{L}_t \rangle - 2\theta$$

(where we used a bound on the spectral norm of  $\mathbf{L}_t$ ), setting  $\theta = 1/T$  gives:

$$\sum_{t=1}^T \mathbb{E} \left[ \langle \mathbf{w}_t \mathbf{w}_t - \mathbf{U}, \mathbf{L}_t \rangle \right] \leq \frac{d \log T}{\eta} + (1 - \gamma) \sum_{t=1}^T \mathbb{E} \left[ \langle \mathbf{W}_t - \tilde{\mathbf{W}}_{t+1}, \tilde{\mathbf{L}}_t \rangle \right] + 2\gamma T + 2.$$

■

#### A.4. The proof of Lemma 10

Let  $\mathbf{W}_t = \sum_{i=1}^d \lambda_i \mathbf{u}_i \mathbf{u}_i^\top$  be the eigendecomposition of  $\mathbf{W}_t$ . Note that due to the assumption that  $\mathbf{L}_t$  has spectral norm bounded by 1,  $|\ell_t| = |\operatorname{tr}(\mathbf{L}_t \mathbf{w}_t \mathbf{w}_t^\top)| \leq \|\mathbf{L}_t\|_\infty \operatorname{tr}(\mathbf{w}_t \mathbf{w}_t^\top) \leq 1$ . We prove the bound separately for the two cases corresponding to the different values of  $B$ .

**On-diagonal sampling** ( $B = 1$ ). When  $B = 1$ , we have

$$\tilde{\mathbf{L}}_t = 2\ell_t \mathbf{W}_t^{-1/2} \mathbf{u}_i \mathbf{u}_i^\top \mathbf{W}_t^{-1/2},$$

for some  $i \in \{1, \dots, d\}$ , so that  $\mathbf{B}_t = 2\ell_t \mathbf{u}_i \mathbf{u}_i^\top$  is rank-one, with single nonzero eigenvalue  $b_{t,1} = 2\ell_t$ . Using (3) gives

$$\langle \mathbf{W}_t - \tilde{\mathbf{W}}_t, \tilde{\mathbf{L}}_t \rangle = \frac{4\eta \ell_t^2}{1 + 2\eta \ell_t},$$

and the claimed result follows by noticing that our assumption on  $\eta$  guarantees  $|\eta \ell_t| \leq \frac{1}{2d} \leq \frac{1}{4}$ .

**Off-diagonal sampling** ( $B = 0$ ). We now have

$$\tilde{\mathbf{L}}_t = \ell_t (\mathbf{W}_t^{-1} \mathbf{w}_t \mathbf{w}_t^\top \mathbf{W}_t^{-1} - \mathbf{W}_t^{-1}), \quad \text{where } \mathbf{w}_t = \sum_{i=1}^d s_i \sqrt{\lambda_i} \mathbf{u}_i.$$

Denoting  $\mathbf{v} = \mathbf{W}_t^{-1/2} \mathbf{w}_t = \sum_{i=1}^d s_i \mathbf{u}_i$ , we get

$$\mathbf{B}_t = \mathbf{W}_t^{1/2} \tilde{\mathbf{L}}_t \mathbf{W}_t^{1/2} = \ell_t (\mathbf{v} \mathbf{v}^\top - \mathbf{I})$$

Using orthonormality of  $\{\mathbf{u}_i\}_{i=1}^T$  we have  $\|\mathbf{v}\|^2 = \sum_i s_i^2 = d$ , which means that  $\mathbf{B}_t$  has a single eigenvalue  $\ell_t(d-1)$ , with the remaining  $d-1$  eigenvalues all equal to  $-\ell_t$ . Using (3):

$$\langle \mathbf{W}_t - \widetilde{\mathbf{W}}_t, \tilde{\mathbf{L}}_t \rangle = \frac{\eta \ell_t^2 (d-1)^2}{1 + \eta \ell_t (d-1)} + (d-1) \frac{\eta \ell_t^2}{1 - \ell_t} \leq 2\eta \ell_t^2 ((d-1)^2 + (d-1)) \leq 2\eta \ell_t^2 d^2,$$

where in the last step we used our assumption on  $\eta$  that ensures both  $|\eta \ell_t| \leq \frac{1}{2}$  and  $|\eta(d-1)\ell_t| \leq \frac{1}{2}$ . This concludes the proof.  $\blacksquare$

### A.5. The proof of Lemma 11

Let  $\mathbf{W}_t = \sum_i \lambda_i \mathbf{u}_i \mathbf{u}_i^\top$  be the eigendecomposition of  $\mathbf{W}_t$ . Since  $\gamma > 0$  the algorithm sample from matrix  $\mathbf{V} = (1-\gamma)\mathbf{W}_t + \frac{\gamma}{d}\mathbf{I}$ , which has the same eigenvectors as  $\mathbf{W}_t$ , and eigenvalues  $\mu_i = (1-\gamma)\lambda_i + \gamma/d$ . Sparse sampling draws indices  $I$  and  $J$  independently from  $\mu$ .

Assume the event  $I = J = i$  occurred with probability  $\mu_i^2$ , for which  $\tilde{\mathbf{L}}_t = \frac{\ell_{ii} \lambda_i}{\mu_i^2} \mathbf{u}_i \mathbf{u}_i^\top$ , with  $\ell_{ii} = \text{tr}(\mathbf{L}_t \mathbf{u}_i \mathbf{u}_i^\top)$ . This means that  $\mathbf{B}_t = \mathbf{W}_t^{1/2} \tilde{\mathbf{L}}_t \mathbf{W}_t^{1/2} = \frac{\ell_{ii} \lambda_i}{\mu_i^2} \mathbf{u}_i \mathbf{u}_i^\top$  has single non-zero eigenvalue  $b_{t,1} = \frac{\ell_{ii} \lambda_i}{\mu_i^2}$ . Using  $(a+b)^2 \geq 4ab$  we have  $\mu_i^2 \geq 4(1-\gamma)\gamma\lambda_i/d \geq 2\gamma\lambda_i/d$ , where we used  $\gamma \leq \frac{1}{2}$  which follows from our assumptions. This implies  $|b_{t,1}| \leq \frac{|\ell_{ii}|d}{2\gamma} \leq \frac{1}{2\eta}$ , which by (3) gives

$$\langle \mathbf{W}_t - \widetilde{\mathbf{W}}_t, \tilde{\mathbf{L}}_t \rangle = \frac{\eta b_{t,1}^2}{1 + \eta b_{t,1}} \leq 2\eta b_{t,1}^2 = 2\eta \frac{\ell_{ii}^2 \lambda_i^2}{\mu_i^4}. \quad (6)$$

Now assume event  $I = i \neq j = J$  occurred with probability  $\mu_i \mu_j$ , for which  $\tilde{\mathbf{L}}_t = \frac{s \ell_{ij}}{2\mu_i \mu_j} (\mathbf{u}_i \mathbf{u}_j^\top + \mathbf{u}_j \mathbf{u}_i^\top)$  with  $\ell_{ij} = \frac{1}{2} \text{tr}(\mathbf{L}_t (\mathbf{u}_i + s\mathbf{u}_j)(\mathbf{u}_i + s\mathbf{u}_j)^\top)$ , where  $s$  is a random sign. This means that  $\mathbf{B}_t = \frac{s \ell_{ij} \sqrt{\lambda_i \lambda_j}}{2\mu_i \mu_j} (\mathbf{u}_i \mathbf{u}_j^\top + \mathbf{u}_j \mathbf{u}_i^\top)$  has two nonzero eigenvalues equal  $b_{t,\pm} = \pm s \frac{\ell_{ij} \sqrt{\lambda_i \lambda_j}}{2\mu_i \mu_j}$ . Using the previously derived bound  $\mu_i^2 \geq 2\gamma\lambda_i/d$ , we have  $|b_{t,\pm}| \leq \frac{\sqrt{\lambda_i \lambda_j}}{2\sqrt{4\gamma^2 \lambda_i \lambda_j / d^2}} = \frac{1}{4\gamma/d} = \frac{1}{4\eta} \leq \frac{1}{2\eta}$ , which, similarly as in (6), implies

$$\langle \mathbf{W}_t - \widetilde{\mathbf{W}}_t, \tilde{\mathbf{L}}_t \rangle \leq 2\eta b_{t,+}^2 + 2\eta b_{t,-}^2 \leq 2\eta \frac{\ell_{ij}^2 \lambda_i \lambda_j}{\mu_i^2 \mu_j^2}. \quad (7)$$

Taking conditional expectation and using (6) and (7) then gives

$$\mathbb{E}[\langle \mathbf{W}_t - \widetilde{\mathbf{W}}_t, \tilde{\mathbf{L}}_t \rangle] \leq 2\eta \sum_{ij} \mu_i \mu_j \frac{\mathbb{E}_s[\ell_{ii}^2] \lambda_i \lambda_j}{\mu_i^2 \mu_j^2} = 2\eta \sum_{ij} \frac{\mathbb{E}_s[\ell_{ii}^2] \lambda_i \lambda_j}{\mu_i \mu_j} \leq 8\eta \sum_{ij} \mathbb{E}_s[\ell_{ii}^2],$$

where  $\mathbb{E}_s[\cdot]$  is the remaining randomization over the sign, and in the last inequality we used  $\frac{\lambda_i}{\mu_i} = \frac{\lambda_i}{(1-\gamma)\lambda_i + \gamma/d} \leq \frac{\lambda_i}{(1-\gamma)\lambda_i} = \frac{1}{1-\gamma} \leq 2$  (because  $\gamma \leq \frac{1}{2}$ ). For the final step of the proof, let us recall the notation  $L_{ij} = \mathbf{u}_i^\top \mathbf{L} \mathbf{u}_j$  and notice that  $\ell_{ii}^2 = L_{ii}^2$ , whereas for  $i \neq j$ :

$$\mathbb{E}_s[\ell_{ij}^2] = \frac{1}{4} \mathbb{E}_s[(L_{ii} + 2sL_{ij} + L_{jj})^2] = \frac{1}{4} \left( (L_{ii} + L_{jj})^2 + 4L_{ij}^2 \right) \leq \frac{1}{4} (2L_{ii}^2 + 2L_{jj}^2 + 4L_{ij}^2)$$

where in the second equality we used  $\mathbb{E}_s[s] = 0$  (so that the cross-terms disappear), while in the last inequality we used  $(a + b)^2 \leq 2a^2 + 2b^2$ . Thus, we obtained

$$\sum_{ij} \mathbb{E}_s [\ell_{ij}^2] \leq d \sum_i L_{ii}^2 + \sum_{i \neq j} L_{ij}^2 \leq d \sum_{ij} L_{ij}^2 = d \|\mathbf{L}\|_F^2,$$

thus proving the statement of the lemma. ■

### A.6. The proof of Corollary 5

From the non-negativity and boundedness of the loss matrices it follows that  $\ell_t = \text{tr}(\mathbf{w}_t \mathbf{w}_t^\top \mathbf{L}_t) \in [0, 1]$ , which implies  $\ell_t^2 \leq \ell_t$ . Let  $L_T^* = \min_{\mathbf{u}: \|\mathbf{u}\|=1} \mathbb{E}[\sum_{t=1}^T \langle \mathbf{u} \mathbf{u}^\top, \mathbf{L}_t \rangle] \leq \bar{L}_T^*$  be the expected loss of the optimal comparator, and let  $\hat{L}_T = \mathbb{E}[\sum_{t=1}^T \ell_t]$  be the algorithm's expected cumulative loss. By Theorem 3 (using  $\ell_t^2 \leq \ell_t$ ):

$$\text{regret}_T = \hat{L}_T - L_T^* \leq \frac{d \log T}{\eta} + \eta(d^2 + 1)\hat{L}_T + 2$$

which can be reordered to imply the bound for  $\eta < 1/(d^2 + 1)$ :

$$(1 - \eta(d^2 + 1))\text{regret}_T \leq \frac{d \log T}{\eta} + \eta(d^2 + 1)\bar{L}_T^* + 2$$

Thus, if  $\bar{L}_T^* \geq 16d^3 \log T$ , we can set  $\eta = \sqrt{\frac{\log T}{d\bar{L}_T^*}} \leq \frac{1}{2(d^2+1)}$  and obtain the bound

$$\text{regret}_T \leq 6d^{3/2} \sqrt{\bar{L}_T^* \log T} + 2$$

Otherwise, we can set  $\eta = 1/(2(d^2 + 1))$  and get

$$\text{regret}_T \leq 24d^3 \log T + 4. \quad \blacksquare$$

### A.7. The proof of Theorem 8

In this section, we provide the proof of our lower bound presented in Theorem 8. Our overall proof strategy is based on the classical recipe for proving worst-case lower bounds in bandit problems—see, e.g., Theorem 5.1 in [Auer et al. \(2002\)](#) or Theorem 6.11 in [Cesa-Bianchi and Lugosi \(2006\)](#). Specifically, we will construct a stochastic adversary and show a lower bound on the regret of any deterministic learning algorithm on this instance, which implies a lower bound on randomized algorithms on any problem instance by Yao's minimax principle ([Yao, 1977](#)). The lower bound for deterministic strategies will be proven using classic information-theoretic arguments. The adversary's strategy will be to draw  $\mathbf{u} \in \mathbb{R}^d$  uniformly at random from the unit sphere before the first round of the game, and play with loss matrices of the form

$$\mathbf{L}_t = Z_t \mathbf{I} - \epsilon \mathbf{u} \mathbf{u}^\top,$$

where  $Z_t \sim N(0, 1)$  and  $\epsilon \in [0, 1]$  is a tuning parameter that will be chosen later. An important feature of this construction is that it keeps the signal-to-noise ratio small by correlating the losses of each action through the global loss  $Z_t$  suffered by each action. This technique is inspired by the work of [Cohen et al. \(2017\)](#), and is crucially important for obtaining a linear scaling with  $d$  in our lower bound.

Note that spectral norm of  $L_t$  is not bounded, but has sub-Gaussian tails. This, however, comes (almost) without loss of generality: by Theorem 7 from [Shamir \(2015\)](#) the lower bound for such sub-Gaussian losses can be converted into a lower bound on the bounded losses at a cost of mere  $\sqrt{\log T}$ .

Define  $\mathbb{E}_{\mathbf{u}}[\cdot] = \mathbb{E}[\cdot | \mathbf{u}]$  as the expectation conditioned on  $\mathbf{u}$  and  $\mathbb{E}_0[\cdot]$  as the total expectation when  $\epsilon = 0$ . Observe that we have  $\mathbb{E}_{\mathbf{u}}[L_t] = -\epsilon \mathbf{u} \mathbf{u}^\top$ , so we can bound the loss of the comparator as

$$\mathbb{E} \left[ \mathbf{U} : \inf_{\mathbf{U}} \sum_{t=1}^T \text{tr}(\mathbf{U} L_t) \right] \leq \mathbb{E} \left[ \mathbb{E}_{\mathbf{u}} \left[ \sum_{t=1}^T \text{tr}(\mathbf{u} \mathbf{u}^\top L_t) \right] \right] = -\epsilon T,$$

where we defined  $\mathbb{E}_{\mathbf{u}}[\cdot] = \mathbb{E}[\cdot | \mathbf{u}]$  as the expectation conditioned on  $\mathbf{u}$ . On the other hand, the expected loss of the learner is given by

$$\mathbb{E} \left[ \sum_{t=1}^T \text{tr}(\mathbf{w}_t \mathbf{w}_t^\top L_t) \right] = -\epsilon \mathbb{E} \left[ \mathbb{E}_{\mathbf{u}} \left[ \sum_{t=1}^T \text{tr}(\mathbf{w}_t \mathbf{w}_t^\top \mathbf{u} \mathbf{u}^\top) \right] \right],$$

so the regret can be lower-bounded as

$$\text{regret}_T \geq \epsilon T - \epsilon \mathbb{E} \left[ \mathbb{E}_{\mathbf{u}} \left[ \sum_{t=1}^T \text{tr}(\mathbf{w}_t \mathbf{w}_t^\top \mathbf{u} \mathbf{u}^\top) \right] \right].$$

Now note that

$$\mathbb{E} \left[ \mathbb{E}_0 \left[ \sum_{t=1}^T \text{tr}(\mathbf{w}_t \mathbf{w}_t^\top \mathbf{u} \mathbf{u}^\top) \right] \right] = \mathbb{E}_0 \left[ \sum_{t=1}^T \text{tr}(\mathbf{w}_t \mathbf{w}_t^\top \mathbb{E}[\mathbf{u} \mathbf{u}^\top]) \right] = \mathbb{E}_0 \left[ \sum_{t=1}^T \text{tr} \left( \mathbf{w}_t \mathbf{w}_t^\top \frac{\mathbf{I}}{d} \right) \right] = \frac{T}{d},$$

where we used the fact that  $\mathbf{u}$  is independent of  $\mathbf{w}_1, \dots, \mathbf{w}_T$  when  $\epsilon = 0$ , and that  $\mathbb{E}[\mathbf{u} \mathbf{u}^\top] = \frac{\mathbf{I}}{d}$  when  $\mathbf{u}$  is uniformly distributed over the unit sphere. Thus, the regret can be rewritten as

$$\text{regret}_T \geq \epsilon T \left( 1 - \frac{1}{d} \right) - \underbrace{\epsilon \mathbb{E} \left[ \mathbb{E}_{\mathbf{u}} \left[ \sum_{t=1}^T \text{tr}(\mathbf{w}_t \mathbf{w}_t^\top \mathbf{u} \mathbf{u}^\top) \right] - \mathbb{E}_0 \left[ \sum_{t=1}^T \text{tr}(\mathbf{w}_t \mathbf{w}_t^\top \mathbf{u} \mathbf{u}^\top) \right] \right]}_{=\Delta_{\mathbf{u}}},$$

which leaves us with the problem of upper-bounding  $\Delta_{\mathbf{u}}$ .

To this end, let  $\ell^T = (\ell_1, \dots, \ell_T)$  be the sequence of losses generated by the deterministic strategy, and let  $p_{\mathbf{u}}(\ell^T)$  denote the density of  $\ell^T$  conditionally on  $\mathbf{u}$ . Notice that  $\mathbf{w}_t$  is completely determined by  $g^{t-1}$ . Furthermore, let  $p_0(g^T)$  denote the corresponding density of  $\ell^T$  when  $\epsilon = 0$ , implying that  $L_t = Z_t \mathbf{I}$  for all  $t$ . Defining  $F(g^T) = \sum_{t=1}^T \text{tr}(\mathbf{w}_t \mathbf{w}_t^\top \mathbf{u} \mathbf{u}^\top)$ , we can write  $\Delta_{\mathbf{u}}$  as

$$\begin{aligned} \Delta_{\mathbf{u}} &= \int F(\ell^T) (p_{\mathbf{u}}(\ell^T) - p_0(\ell^T)) \, d\ell^T \leq \int_{p_{\mathbf{u}}(\ell^T) \geq p_0(\ell^T)} F(\ell^T) (p_{\mathbf{u}}(\ell^T) - p_0(\ell^T)) \, d\ell^T \\ &\leq T \int_{p_{\mathbf{u}}(\ell^T) \geq p_0(\ell^T)} (p_{\mathbf{u}}(\ell^T) - p_0(\ell^T)) \, d\ell^T \leq T D_{\text{TV}}(p_0 \| p_{\mathbf{u}}) \leq T \sqrt{\frac{1}{2} D_{\text{KL}}(p_0 \| p_{\mathbf{u}})}, \end{aligned}$$



where  $D_{\text{TV}}(\cdot\|\cdot)$  and  $D_{\text{KL}}(\cdot\|\cdot)$  denote, respectively, the total variation distance and the Kullback-Leibler (KL) divergence between two distributions, and the last step uses Pinsker's inequality, while the second inequality uses  $F(g^T) = \sum_t (\mathbf{w}_t^\top \mathbf{u})^2 \leq T$ . By the chain rule for the KL divergence, we have

$$D_{\text{KL}}(p_0\|p_{\mathbf{u}}) = \sum_{t=1}^T \mathbb{E}_0 [D_{\text{KL}}(p_0(\ell_t|\ell^{t-1})\|p_{\mathbf{u}}(\ell_t|\ell^{t-1}))]$$

Now, the loss in round  $t$  can be written as  $\ell_t = \mathbf{w}_t^\top \mathbf{L}_t \mathbf{w}_t$ . By the definition of  $\mathbf{L}_t$ , the conditional distribution of  $\ell_t$  is Gaussian with unit variance under both  $p_{\mathbf{u}}$  and  $p_0$ :  $\ell_t = Z_t - \epsilon(\mathbf{w}_t^\top \mathbf{u})^2 \sim N(-\epsilon(\mathbf{w}_t^\top \mathbf{u})^2, 1)$  under  $p_{\mathbf{u}}$  and  $\ell_t = Z_t \sim N(0, 1)$  under  $p_0$ . Thus, the conditional KL divergence between the two distributions can be written as

$$D_{\text{KL}}(p_0(g_t|g^{t-1})\|p_{\mathbf{u}}(g_t|g^{t-1})) = \frac{1}{2}\epsilon^2(\mathbf{w}_t^\top \mathbf{u})^4,$$

which implies

$$\Delta_{\mathbf{u}} \leq \frac{T}{2}\epsilon \sqrt{\sum_{t=1}^T \mathbb{E}_0 [(\mathbf{w}_t^\top \mathbf{u})^4]}.$$

In order to bound  $\mathbb{E}[\Delta_{\mathbf{u}}]$ , we use Jensen's inequality  $\mathbb{E}[\sqrt{\cdot}] \leq \sqrt{\mathbb{E}[\cdot]}$  to write

$$\mathbb{E}[\Delta_{\mathbf{u}}] \leq \frac{T}{2}\epsilon \sqrt{\sum_{t=1}^T \mathbb{E}[\mathbb{E}_0 [(\mathbf{w}_t^\top \mathbf{u})^4]]} = \frac{T}{2}\epsilon \sqrt{\sum_{t=1}^T \mathbb{E}_0 [\mathbb{E} [(\mathbf{w}_t^\top \mathbf{u})^4]]},$$

where in the last step we swapped the order of expectations as  $\mathbf{u}$  is independent of  $\ell_1, \dots, \ell_T$  under  $p_0$ . Since  $\mathbf{u}$  is distributed uniformly over the unit sphere,  $(\mathbf{w}_t^\top \mathbf{u})$  has the same distribution as  $u_1$ . Using the fact that  $u_1^2 \sim \text{Beta}(\frac{1}{2}, \frac{d-1}{2})$  (Devroye, 1986), this implies:

$$\mathbb{E}[(\mathbf{w}_t^\top \mathbf{u})^4] = \mathbb{E}[u_1^4] = \frac{3}{d(d+2)},$$

Thus, we arrive to the bound

$$\mathbb{E}[\Delta_{\mathbf{u}}] \leq \frac{T}{2}\epsilon \sqrt{T \frac{3}{d(d+2)}} \leq \frac{T^{3/2}}{d}\epsilon,$$

which, put together with the previous calculations, eventually gives

$$\text{regret}_T \geq \epsilon T \left(1 - \frac{1}{d}\right) - \epsilon^2 \frac{T^{3/2}}{d}.$$

Bounding  $1 - \frac{1}{d} \geq \frac{1}{2}$  and setting  $\epsilon = dT^{-1/2}/4$  gives  $\text{regret}_T = \Omega(d\sqrt{T})$ , which by aforementioned Theorem 7 from Shamir (2015) implies the claim in the theorem.  $\blacksquare$

## Appendix B. Efficient implementation of the update

In this section, we give details on the efficient implementation of the mirror descent update (1):

$$\begin{aligned} \text{(update step)} \quad \widetilde{\mathbf{W}}_{t+1} &= \underset{\mathbf{W}}{\operatorname{argmin}} \left\{ D_R(\mathbf{W} \parallel \mathbf{W}_t) + \eta \operatorname{tr}(\mathbf{W} \widetilde{\mathbf{L}}_t) \right\}, \\ \text{(projection step)} \quad \mathbf{W}_{t+1} &= \underset{\mathbf{W} \in \mathcal{W}}{\operatorname{argmin}} D_R(\mathbf{W} \parallel \widetilde{\mathbf{W}}_{t+1}), \end{aligned}$$

with the Bregman divergence induced by the negative log-determinant regularizer:

$$D_R(\mathbf{W} \parallel \mathbf{U}) = \operatorname{tr}(\mathbf{U}^{-1} \mathbf{W}) - \log \frac{\det(\mathbf{W})}{\det(\mathbf{U})} - d$$

As we will show, the algorithm runs in time  $\tilde{\mathcal{O}}(d)$  per trial for sparse sampling method, and in time  $\tilde{\mathcal{O}}(d^3)$  for dense sampling method. In what follows, we assume that the eigenvalue decomposition  $\mathbf{W}_t = \sum_i \mu_i \mathbf{u}_i \mathbf{u}_i^\top$  is given at the beginning of trial  $t$ , where  $\{\mathbf{u}_i\}_{i=1}^d$  are the eigenvectors, and  $\{\lambda_i\}_{i=1}^d$  are the eigenvalues of  $\mathbf{W}_t$  (sorted in a decreasing order), and we dropped the trial index for the sake of clarity. The eigenvalues of  $\mathbf{W}_t$  then get mixed with a uniform distribution:

$$\lambda_i = (1 - \gamma) \mu_i + \gamma \frac{1}{d}, \quad i = 1, \dots, d$$

(with  $\gamma = 0$  for dense sampling) and are used to sample the action of the algorithm.

### B.1. The update step

We have shown in Section 4 that the unprojected solution is given by (2):

$$\widetilde{\mathbf{W}}_{t+1} = \mathbf{W}_t^{1/2} (\mathbf{I} + \eta \mathbf{B}_t)^{-1} \mathbf{W}_t^{1/2}, \quad \text{where } \mathbf{B}_t = \mathbf{W}_t^{1/2} \widetilde{\mathbf{L}}_t \mathbf{W}_t^{1/2}.$$

**Sparse sampling.** Two indices  $I, J \in \{1, \dots, d\}$  are independently sampled from the same distribution satisfying  $\mathbb{P}[J = i] = \mathbb{P}[I = i] = \lambda_i$  (which takes negligible  $\mathcal{O}(\log d)$  time).

When  $I = J$ , the algorithm plays with  $\mathbf{w} = \mathbf{u}_I$ , receives  $\ell_t$ , and the loss estimate is given by  $\widetilde{\mathbf{L}} = \frac{\ell_t}{\lambda_I^2} \mathbf{u}_I \mathbf{u}_I^\top$ . As  $\mathbf{u}_I$  is one of the eigenvectors of  $\mathbf{W}_t$ , we obtain  $\mathbf{B}_t = \ell_t \frac{\mu_I}{\lambda_I^2} \mathbf{u}_I \mathbf{u}_I^\top$ . This means that  $\mathbf{W}_t$  and  $\mathbf{I} + \eta \mathbf{B}_t$  commute so that  $\widetilde{\mathbf{W}}_{t+1}$  has the same eigensystem as  $\mathbf{W}_t$  and it only amounts to computing the eigenvalues  $(\mu'_1, \dots, \mu'_d)$  of  $\widetilde{\mathbf{W}}_{t+1}$ , which are given by:

$$\mu'_i = \begin{cases} \mu_i & \text{for } i \neq I, \\ \frac{1}{1 + \eta \ell_t \mu_I / \lambda_I^2} \mu_I & \text{for } i = I. \end{cases}$$

As the eigenvectors do not change, and only one eigenvalue is updated, the eigendecomposition of  $\widetilde{\mathbf{W}}_{t+1}$  is updated in time  $\mathcal{O}(1)$ .

When  $I \neq J$ , the algorithm plays with  $\mathbf{w} = \frac{1}{\sqrt{2}}(\mathbf{u}_I + s \mathbf{u}_J)$ , where  $s \in \{-1, 1\}$  is a random sign. The loss estimate is  $\widetilde{\mathbf{L}} = \frac{s \ell_t}{2 \lambda_I \lambda_J} (\mathbf{u}_I \mathbf{u}_J^\top + \mathbf{u}_J \mathbf{u}_I^\top)$ , which gives  $\mathbf{B}_t = \frac{\sqrt{\mu_I \mu_J} s \ell_t}{2 \lambda_I \lambda_J} (\mathbf{u}_I \mathbf{u}_J^\top + \mathbf{u}_J \mathbf{u}_I^\top)$ . To simplify notation, we denote:

$$\mathbf{I} + \eta \mathbf{B}_t = \mathbf{I} + \beta (\mathbf{u}_I \mathbf{u}_J^\top + \mathbf{u}_J \mathbf{u}_I^\top), \quad \text{where } \beta = \frac{\eta \sqrt{\mu_I \mu_J} s \ell_t}{2 \lambda_I \lambda_J}.$$

Due to rank-two representation of  $\mathbf{B}_t$ , which involves only two eigenvectors of  $\mathbf{W}_t$ , the eigenvectors and eigenvalues of  $\widetilde{\mathbf{W}}_{t+1}$  will be the same as for  $\mathbf{W}_t$ , except for those associated with the drawn indices  $I$  and  $J$ . Specifically, it can be verified by a direct computation that the inverse of  $\mathbf{I} + \eta\mathbf{B}_t$  is given by:

$$(\mathbf{I} + \beta(\mathbf{u}_I\mathbf{u}_J^\top + \mathbf{u}_J\mathbf{u}_I^\top))^{-1} = \mathbf{I} + \frac{\beta^2}{1 - \beta^2}(\mathbf{u}_I\mathbf{u}_I^\top + \mathbf{u}_J\mathbf{u}_J^\top) - \frac{\beta}{1 - \beta^2}(\mathbf{u}_I\mathbf{u}_J^\top + \mathbf{u}_J\mathbf{u}_I^\top).$$

Multiplying the above from both sides by  $\mathbf{W}_t^{1/2}$  gives:

$$\begin{aligned} \widetilde{\mathbf{W}}_{t+1} &= \mathbf{W}_t + \frac{\beta^2}{1 - \beta^2}(\mu_I\mathbf{u}_I\mathbf{u}_I^\top + \mu_J\mathbf{u}_J\mathbf{u}_J^\top) - \frac{\beta\sqrt{\mu_I\mu_J}}{1 - \beta^2}(\mathbf{u}_I\mathbf{u}_J^\top + \mathbf{u}_J\mathbf{u}_I^\top) \\ &= \sum_{i \notin \{I, J\}} \mu_i\mathbf{u}_i\mathbf{u}_i^\top + \frac{1}{1 - \beta^2} \left( \mu_I\mathbf{u}_I\mathbf{u}_I^\top + \mu_J\mathbf{u}_J\mathbf{u}_J^\top - \beta\sqrt{\mu_I\mu_J}(\mathbf{u}_I\mathbf{u}_J^\top + \mathbf{u}_J\mathbf{u}_I^\top) \right). \end{aligned}$$

As the term in parentheses on the right-hand side only concerns the subspace spanned by  $\mathbf{u}_I$  and  $\mathbf{u}_J$ ,  $\widetilde{\mathbf{W}}_{t+1}$  has eigendecomposition  $\widetilde{\mathbf{W}}_{t+1} = \sum_{i \notin \{I, J\}} \mu_i\mathbf{u}_i\mathbf{u}_i^\top + \mu_+\mathbf{u}_+\mathbf{u}_+^\top + \mu_-\mathbf{u}_-\mathbf{u}_-^\top$ , where  $\mathbf{u}_+$  and  $\mathbf{u}_-$  are linear combinations of  $\mathbf{u}_I$  and  $\mathbf{u}_J$ . Specifically:

$$\begin{aligned} \mu_\pm &= \frac{\mu_I + \mu_J \pm \sqrt{(\mu_I - \mu_J)^2 + 4\mu_I\mu_J\beta^2}}{2(1 - \beta^2)}, \\ \mathbf{u}_\pm &= \frac{-\beta\sqrt{\mu_I\mu_J}\mathbf{u}_I + (\mu_\pm(1 - \beta^2) - \mu_I)\mathbf{u}_J}{\sqrt{\beta^2\mu_I\mu_J + (\mu_\pm(1 - \beta^2) - \mu_I)^2}}. \end{aligned}$$

Thus, we only need to update two eigenvalues and their corresponding eigenvectors, which can be done in  $O(d)$ .

**Dense sampling.** For the ‘‘on-diagonal’’ sampling,  $\mathbf{B}_t = 2\ell_t\mathbf{u}_i\mathbf{u}_i^\top$ , where  $\mathbf{u}_i$  is one of the eigenvectors of  $\mathbf{W}_t$ . This means that  $\mathbf{W}_t$  and  $\mathbf{I} + \eta\mathbf{B}_t$  commute so that  $\widetilde{\mathbf{W}}_{t+1}$  has the same eigensystem as  $\mathbf{W}_t$  and it only amounts to computing the eigenvalues  $(\lambda'_1, \dots, \lambda'_d)$  of  $\widetilde{\mathbf{W}}_{t+1}$ , which are given by:

$$\lambda'_j = \begin{cases} \lambda_j & \text{for } j \neq i \\ \frac{1}{1+2\eta\ell_t}\lambda_i & \text{for } j = i \end{cases}$$

For the ‘‘off-diagonal’’ sampling, we have  $\mathbf{B}_t = \ell_t(\mathbf{v}_t\mathbf{v}_t^\top - \mathbf{I})$  where  $\mathbf{v}_t = \sum_{i=1}^d s_i\mathbf{u}_i$ . Using Sherman-Morrison formula we can invert  $\mathbf{I} + \eta\mathbf{B}_t = \mathbf{I}(1 - \eta\ell_t) + \eta\ell_t\mathbf{v}_t\mathbf{v}_t^\top$  to get:

$$\widetilde{\mathbf{W}}_{t+1} = \frac{1}{1 - \eta\ell_t}\mathbf{W}_t^{1/2} \left( \mathbf{I} - \frac{\eta\ell_t\mathbf{v}_t\mathbf{v}_t^\top}{1 + \eta(d-1)\ell_t} \right) \mathbf{W}_t^{1/2},$$

where we used  $\mathbf{v}_t^\top\mathbf{v}_t = \|\mathbf{v}_t\|^2 = d$ . To calculate the eigendecomposition of  $\widetilde{\mathbf{W}}_{t+1}$ , we rewrite the expression above as:

$$\widetilde{\mathbf{W}}_{t+1} = \frac{1}{1 - \eta\ell_t}\mathbf{U} \underbrace{\left( \mathbf{\Lambda} - \frac{\eta\ell_t\tilde{\mathbf{v}}_t\tilde{\mathbf{v}}_t^\top}{1 + \eta(d-1)\ell_t} \right)}_{\mathbf{A}} \mathbf{U}^\top,$$

where  $\mathbf{U} = [\mathbf{u}_1, \dots, \mathbf{u}_d]$  stores the eigenvectors of  $\mathbf{W}_t$  as columns,  $\mathbf{\Lambda} = \text{diag}(\lambda_1, \dots, \lambda_d)$ , and  $\tilde{\mathbf{v}}_t = \sum_{i=1}^d s_i \lambda_i^{1/2} \mathbf{e}_i$ , with  $\mathbf{e}_i$  being the  $i$ -th unit vector (with  $i$ -th coordinate equal to 1 and remaining coordinates equal to 0). Thus, we first calculate the eigendecomposition of  $\mathbf{A}$ , and then multiply the resulting eigenvectors by  $\mathbf{U}$  to get the eigendecomposition of  $\widetilde{\mathbf{W}}_{t+1}$ . We note that  $\mathbf{A}$  is a rank-one update of the diagonal matrix, which eigendecomposition can be calculated in  $\mathcal{O}(d^2)$  (Gu and Eisenstat, 1994). However, the multiplication of eigenvectors of  $\mathbf{A}$  by  $\mathbf{U}$  still takes  $\mathcal{O}(d^3)$ , which is also the dominating cost of the whole update with dense sampling.

## B.2. The projection step

The projection step reduces to solving:

$$\mathbf{W}_t = \underset{\mathbf{W} \in \mathcal{W}}{\text{argmin}} \text{tr} \left( \widetilde{\mathbf{W}}_{t+1}^{-1} \mathbf{W} \right) - \log \det(\mathbf{W}). \quad (8)$$

We first argue that  $\mathbf{W}_t$  and  $\widetilde{\mathbf{W}}_{t+1}$  have the same eigenvectors, and the projection only affects the eigenvalues. Note that  $\det(\mathbf{W})$  only depends on the eigenvalues of  $\mathbf{W}$  and not on its eigenvectors. Furthermore, for any symmetric matrices  $\mathbf{A}$  and  $\mathbf{B}$ ,  $\text{tr}(\mathbf{A}\mathbf{B}) \geq \sum_{i=1}^d \lambda_{d-i}(\mathbf{A})\lambda_i(\mathbf{B})$ , where  $\lambda_i(\mathbf{A})$ ,  $\lambda_i(\mathbf{B})$  denote the eigenvalues of  $\mathbf{A}$  and  $\mathbf{B}$ , respectively, sorted in a decreasing order (Bernstein, 2009, Fact 5.12.4). This means that if we let  $\boldsymbol{\nu} = (\nu_1, \dots, \nu_d)$  and  $\boldsymbol{\mu} = (\mu_1, \dots, \mu_d)$  denote the eigenvalues of  $\widetilde{\mathbf{W}}_{t+1}$  and  $\mathbf{W}$ , respectively, sorted in a decreasing order, then  $\text{tr}(\widetilde{\mathbf{W}}_{t+1}^{-1} \mathbf{W}) \geq \sum_{i=1}^d \nu_i^{-1} \mu_i$ , with the equality if and only if the eigenvectors of  $\widetilde{\mathbf{W}}_{t+1}$  and  $\mathbf{W}^{-1}$  are the same. This means that if we fix the eigenvalues of  $\mathbf{W}$ , then the right-hand side of (8) is minimized by  $\widetilde{\mathbf{W}}_{t+1}$  and  $\mathbf{W}_t$  sharing their eigenvectors.

Thus, the projection can be reduced to finding the eigenvalues  $\boldsymbol{\mu}$  of  $\mathbf{W}_t$ :

$$\boldsymbol{\mu} = \underset{\boldsymbol{\mu} \in \mathcal{M}}{\text{argmin}} \sum_{i=1}^d \frac{\mu_i}{\nu_i} - \log \mu_i, \quad \mathcal{M} = \{\boldsymbol{\mu}: \mu_1 \geq \mu_2 \geq \dots \geq \mu_d \geq 0, \sum_i \mu_i = 1\}$$

In fact, the first constraint in  $\mathcal{M}$  is redundant, as the positivity of  $\mu_i$  is implied by the domain of the logarithmic function, and if  $\mu_i < \mu_{i+1}$  for any  $i$  such that  $\nu_i > \nu_{i+1}$ , then it is straightforward to see that swapping the values of  $\mu_i$  and  $\mu_{i+1}$  decreases the objective function. Taking the derivative of the right-hand side and incorporating the constraint  $\sum_i \mu_i = 1$  by introducing the Lagrange multiplier  $\theta$  gives for any  $i = 1, \dots, d$ :

$$\nu_i^{-1} - \mu_i^{-1} + \theta = 0 \quad \implies \quad \mu_i = \frac{1}{\nu_i^{-1} + \theta}$$

The value of  $\theta$  satisfying  $\sum_i \mu_i = 1$  can now be easily obtained by a root-find algorithm, e.g., by the Newton method (alternatively, we can cast the problem as one-dimensional minimization of a convex function  $f(\theta) = -\sum_i \log(\mu_i^{-1} + \theta) - \theta$ ). As the time complexity of a single iteration is  $\mathcal{O}(d)$  and the number of iterations required to achieve error of order  $\epsilon$  is at most  $\mathcal{O}(\log \epsilon^{-1})$ , the total runtime is  $\mathcal{O}(d \log \epsilon^{-1})$ . Since the errors may generally accumulate over time we need to set  $\epsilon^{-1}$  to scale polynomially with  $T$  (so that the total error at the end of the game will still be negligible), which means that the runtime is of order  $\mathcal{O}(d \log T) = \tilde{\mathcal{O}}(d)$ .

### Appendix C. Matrix Hedge and Tsallis regularizers

In this section we explain some technical difficulties that we faced while attempting to analyze variants of our algorithm based on the regularization functions most commonly used in multi-armed bandit problems: Tsallis entropies and the Shannon entropy (known as the quantum entropy function in the matrix case). This section is not to be regarded as a counterexample against any of these algorithms, but rather a summary of semi-formal arguments suggesting that the algorithms derived from these regularization functions may fail to give near-optimal performance guarantees. In fact, we believe that obstacles we outline here might be impossible to overcome.

**Matrix Hedge.** Consider the online mirror descent algorithm (1) equipped with the quantum negative entropy regularizer  $R(\mathbf{U}) = \text{tr}(\mathbf{U} \log \mathbf{U})$  and any unbiased loss estimate  $\widehat{\mathbf{L}}_t$  satisfying  $|\eta \mathbf{L}_t| = \mathcal{O}(1)$  (which can be achieved by an appropriate amount of forced exploration, without loss of generality). This corresponds to a straightforward bandit variant of the algorithm known as Matrix Hedge (MH) (Tsuda et al., 2005; Arora et al., 2005; Warmuth and Kuzmin, 2008). Following standard derivations (e.g., by Hazan et al., 2017), one can easily show an upper bound on the regret of the form

$$\text{regret}_T \leq \frac{\ln d}{\eta} + c_1 \eta \sum_{t=1}^T \mathbb{E} \left[ \text{tr} \left( \mathbf{W}_t \widetilde{\mathbf{L}}_t^2 \right) \right] + c_2,$$

for some constants  $c_1$  and  $c_2$ . What is thus left is to bound the ‘‘variance’’ terms  $\mathbb{E}[\text{tr}(\mathbf{W}_t \widetilde{\mathbf{L}}_t^2)]$  by a (possibly dimension-dependent) constant for all  $t$ , and tune the learning rate appropriately. While this is easily accomplished in the standard multi-armed bandit setup by exploiting the properties of importance-weighted loss estimates, controlling this term becomes much harder in the matrix case.

We formally show below that the variance term described above cannot be upper bounded by any constant for any natural choice of unbiased loss estimator. In what follows, we drop the time index  $t$  for the sake of clarity. We assume the loss estimate has a general form  $\widetilde{\mathbf{L}} = \ell \mathbf{H}$ , where  $\ell = \langle \mathbf{L}, \mathbf{w} \mathbf{w}^\top \rangle$  is the observed loss and  $\mathbf{H}$  is some matrix that does not depend on  $\ell$  (but will depend on the action  $\mathbf{w} \mathbf{w}^\top$  of the learner). Notably, this class of loss estimators include all known unbiased loss estimators for linear bandits. We will show that when  $\mathbf{L} \succeq \alpha \mathbf{I}$ , then  $\mathbb{E} \left[ \text{tr} \left( \mathbf{W} \widetilde{\mathbf{L}}^2 \right) \right] \geq \frac{c}{\lambda_{\min}(\mathbf{W})}$ , where  $\alpha$  and  $c$  are some positive constants, and  $\lambda_{\min}(\mathbf{W})$  is the smallest eigenvalue of  $\mathbf{W}$ . This clearly implies that one cannot upper bound the variance terms by a constant, since there is no way in general to lower bound  $\lambda_{\min}(\mathbf{W})$  by a constant independent of  $T$ .

To make the analysis as simple as possible, consider the case  $d = 2$  and (without loss of generality) assume  $\mathbf{W} = \text{diag}(\lambda_1, \lambda_2)$ . Let  $\mathbf{w} = (w_1, w_2)$  be the action of the algorithm. Since  $\mathbb{E}[\mathbf{w} \mathbf{w}^\top] = \mathbf{W}$  we have:

$$\mathbb{E}[w_1^2] = \lambda_1, \quad \mathbb{E}[w_2^2] = \lambda_2.$$

Furthermore the observed loss is given by:

$$\ell = \text{tr}(\mathbf{w} \mathbf{w}^\top \mathbf{L}) = \mathbf{w}^\top \mathbf{L} \mathbf{w} = w_1^2 L_{11} + w_2^2 L_{22} + w_1 w_2 L_{12},$$

where  $L_{ij}$  are the entries of  $\mathbf{L}$ . The condition  $\mathbb{E}[\widetilde{\mathbf{L}}] = \mathbf{L}$  thus implies:

$$\mathbb{E}[(w_1^2 L_{11} + 2w_1 w_2 L_{12} + w_2^2 L_{22}) \mathbf{H}_{12}] = L_{12},$$

where  $\mathbf{H}_{12}$  is the off-diagonal entry of  $\mathbf{H}$ . The right-hand side of the above does not depend on  $L_{11}$  and  $L_{22}$ , and since these numbers can be arbitrarily chosen by the adversary, the left-hand side

cannot depend on them either. This means that  $\mathbb{E}[w_1^2 H_{12}] = \mathbb{E}[w_2^2 H_{12}] = 0$ , and  $\mathbb{E}[w_1 w_2 H_{12}] = \frac{1}{2}$ . From the last expression we get:

$$\frac{1}{2} = \mathbb{E}[w_1 w_2 H_{12}] \leq \sqrt{\mathbb{E}[w_1^2 w_2^2]} \sqrt{\mathbb{E}[H_{12}^2]} \implies \mathbb{E}[H_{12}^2] \geq \frac{1}{4\mathbb{E}[w_1^2 w_2^2]} \geq \frac{1}{4 \min\{\lambda_1, \lambda_2\}}$$

where the inequality on the left is Cauchy–Schwarz, while the inequality on the right uses

$$\mathbb{E}[w_1^2 w_2^2] \leq \mathbb{E}[w_1^2] = \lambda_1, \quad \mathbb{E}[w_1^2 w_2^2] \leq \mathbb{E}[w_2^2] = \lambda_2.$$

From the assumption  $\mathbf{L} \succeq \alpha \mathbf{I}$  we have  $\ell \geq \alpha$ , which gives

$$\mathbb{E}[\text{tr}(\mathbf{W}\tilde{\mathbf{L}}^2)] = \mathbb{E}[\text{tr}(\mathbf{W}\ell^2 \mathbf{H}^2)] \geq \alpha^2 \mathbb{E}[\text{tr}(\mathbf{W}\mathbf{H}^2)].$$

Since

$$\begin{aligned} \mathbf{W}\mathbf{H}^2 &= \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} \begin{bmatrix} H_{11} & H_{12} \\ H_{12} & H_{22} \end{bmatrix} \begin{bmatrix} H_{11} & H_{12} \\ H_{12} & H_{22} \end{bmatrix} \\ &= \begin{bmatrix} \lambda_1(H_{11}^2 + H_{12}^2) & \lambda_1(H_{11}H_{12} + H_{12}H_{22}) \\ \lambda_2(H_{11}H_{12} + H_{12}H_{22}) & \lambda_2(H_{22}^2 + H_{12}^2) \end{bmatrix}, \end{aligned}$$

this implies

$$\mathbb{E}[\text{tr}(\mathbf{W}\mathbf{H}^2)] = \lambda_1 \mathbb{E}[H_{11}^2] + \lambda_2 \mathbb{E}[H_{22}^2] + \mathbb{E}[H_{12}^2] \geq \mathbb{E}[H_{12}^2] \geq \frac{1}{4 \min\{\lambda_1, \lambda_2\}},$$

and therefore  $\mathbb{E}[\text{tr}(\mathbf{W}\tilde{\mathbf{L}}^2)] \geq \frac{\gamma^2}{4\lambda_{\min}(\mathbf{W})}$ .

**Tsallis regularizers.** A similar analysis can be done for the case of matrix Tsallis regularizers  $R(\mathbf{U}) = -\text{tr}(\mathbf{U}^\alpha)$  with  $\alpha \in (0, 1)$ , which are related to Tsallis entropy (Abernethy et al., 2015; Allen-Zhu et al., 2015; Allen-Zhu et al., 2017). In this case the variance term  $\text{tr}(\mathbf{W}_t \tilde{\mathbf{L}}_t^2)$  in Matrix Hedge can be replaced by the squared *local norms* of the losses (Shalev-Shwartz, 2011; Lattimore and Szepesvári, 2019; Hazan, 2015), defined as  $\nabla^{-2}R(\mathbf{W}_t)[\tilde{\mathbf{L}}_t, \tilde{\mathbf{L}}_t]$ , where  $\nabla^{-2}R$  is the inverse Hessian of the regularizer. As the Tsallis regularizer is a symmetric spectral function, one can get a closed-form expression for the quadratic form of its Hessian Lewis and Sendov (2002). Employing convex duality (by identifying  $\nabla^{-2}R$  with  $\nabla^2 R^*$ , where  $R^*$  is the convex conjugate of  $R$ ), and lower bounding, one arrives at the following simple bound on the local norm:

$$\nabla^{-2}R(\mathbf{W}_t)[\tilde{\mathbf{L}}_t, \tilde{\mathbf{L}}_t] \geq c \text{tr}(\mathbf{W}_t \tilde{\mathbf{L}}_t \mathbf{W}_t^{1-\alpha} \tilde{\mathbf{L}}_t).$$

It is known that the negative entropy is the  $\alpha \rightarrow 1$  limit of (properly normalized) Tsallis regularizer, while the  $\alpha \rightarrow 0$  limit is the log-determinant regularizer. Interestingly, the expression above indeed turns into the MH variance term  $\text{tr}(\mathbf{W}_t \tilde{\mathbf{L}}_t^2)$  for  $\alpha = 1$ , and to the term  $\text{tr}(\mathbf{B}_t^2)$  with  $\mathbf{B}_t = \mathbf{W}_t^{1/2} \tilde{\mathbf{L}}_t \mathbf{W}_t^{1/2}$  for  $\alpha = 0$ , which we encountered in our proofs (compare with (3) for  $\eta \rightarrow 0$ ).

One can repeat the same arguments as in the case of the MH variance term to obtain the lower bound

$$\mathbb{E}_t \left[ \text{tr}(\mathbf{W}_t \tilde{\mathbf{L}}_t \mathbf{W}_t^{1-\alpha} \tilde{\mathbf{L}}_t) \right] \geq \frac{c}{(\lambda_{\min}(\mathbf{W}_t))^\alpha},$$

as long as the loss estimate is unbiased and has the same general form as in the MH case. This suggests that the only way to control these local norms is to take  $\alpha = 0$ , resulting in the log-determinant regularizer that we use in our main algorithms in the present paper.

We would like to stress one more time that the above arguments do not constitute a lower bound on the performance of these algorithms; we merely lower-bound the terms from which all known upper bounds are derived for linear bandit problems. At best, this suggests that significantly new techniques are required to prove positive results about these algorithms. We ourselves are, however, more pessimistic and believe that these algorithms cannot provide regret guarantees of optimal order.