AdaTask: Adaptive Multitask Online Learning

Pierre Laforgue*
Department of Computer Science
University of Milan, Italy
pierre.laforgue@unimi.it

Andrea Della Vecchia*
MaLGa, DIMA
UniGE, Italy
dellavecchia@dima.unige.it

Nicolò Cesa-Bianchi
Department of Computer Science
University of Milan, Italy
nicolo.cesa-bianchi@unimi.it

Lorenzo Rosasco
MaLGa, DIBRIS, UniGE, Italy
IIT, Italy and CBMM, MIT, USA
lorenzo.rosasco@unige.it

Abstract
We introduce and analyze AdaTask, a multitask online learning algorithm that adapts to the unknown structure of the tasks. When the $N$ tasks are stochastically activated, we show that the regret of AdaTask is better, by a factor that can be as large as $\sqrt{N}$, than the regret achieved by running $N$ independent algorithms, one for each task. AdaTask can be seen as a comparator-adaptive version of Follow-the-Regularized-Leader with a Mahalanobis norm potential. Through a variational formulation of this potential, our analysis reveals how AdaTask jointly learns the tasks and their structure. Experiments supporting our findings are presented.

1 Introduction
The guiding principle of multitask learning (Caruana, 1997) can be formulated as follows: if several tasks are similar, solving them together should be more data efficient than solving them independently. However, understanding whether the tasks are similar or not, in what sense they are similar, and how to exploit this similarity, is often a nontrivial question, almost as hard as learning the tasks themselves. In this paper, we introduce and analyze AdaTask, a multitask online learning algorithm that automatically adapts to the unknown task structure. In contrast to prior approaches, we show that AdaTask is able to adapt to any kind of task structure, while enjoying better regret guarantees than independent learning when the task activations are stochastic.

Many approaches to multitask learning have been proposed in the past. In the batch setting, Laplacian penalization, that favors predictors which are similar according to a given graph over the tasks—but without the possibility of learning the best graph—was first introduced in (Evgeniou et al., 2005). A few years later, a dual approach based on identifying common features was investigated by Argyriou et al. (2008), while Ciliberto et al. (2015) recently revisited these ideas to propose a convex relaxation that allows to learn jointly (via an alternate minimization procedure) the tasks and their structure. Note that these works prove algorithmic convergence, but do not provide any learning guarantees.

In the online setting, Cavallanti et al. (2010) leverage the ideas of Argyriou et al. (2005) to devise a multitask Perceptron algorithm that enjoys improved mistake bounds when the tasks are similar. Cesa-Bianchi et al. (2021) have later extended these results to general Online Mirror Descent, and propose a way to adapt to the task variance. An important limitation of these works however, is that task similarity is only considered through the angle of the task variance, that fails to account for many interesting task structures, see Figure 1. In contrast, Saha et al. (2011) proposed an algorithm for jointly learning the task vectors and their structure, but without providing any regret guarantees.

*equal contribution.

Preprint. Under review.
Cesa-Bianchi et al. (2021) have established a regret bound that depends on $1 + (N - 1)\text{Var}(U)$, where $\text{Var}(U) = 1/(N - 1) \sum_{i=1}^{N} \|u_i - \bar{u}\|_2^2$, with $\bar{u} = (1/N) \sum_{i=1}^{N} u_i$, is the task variance. Instead, we show that the expectation of the regret of AdaTask w.r.t. $U$ scales with $\|U\|_{S(1)}$, the Schatten 1-norm of $U$. In scenario a), where the tasks are all clustered, both descriptions correctly capture the task structure, and yield a $O(\sqrt{T})$ regret bound. However, as soon as the tasks are not uniformly spread, as in scenarios b), the variance characterization fails to account for the task structure and provides a $O(\sqrt{NT})$ bound. Note also that $\|U\|_{S(1)}$ allows to capture negative correlations. More generally, it is possible to show that the Schatten 1-norm characterization is optimal among Mahalanobis metrics, see Equation (6) for details.

From a technical perspective, multitask improvements are usually obtained by variants of known algorithms (e.g., Online Mirror Descent) that use Mahalanobis regularizers, i.e., Euclidean norms weighted by a positive semidefinite matrix. The Mahalanobis matrix, which we refer to as the interaction matrix, induces a form of communication between the tasks, see Equation (2). If the interaction matrix is aligned with the task structure, some improvements are expected. However, standard results from Online Convex Optimization (OCO) Hazan (2016); Orabona (2019) only apply to regularizers with fixed interaction matrices. In other words, the learner commits beforehand to a similarity pattern, which acts as a learning bias, and suffers a regret that depends on how well this scheme is matching the task structure. Ideally, a regret bound should scale with the best possible interaction matrix in hindsight—note that such a matrix always exists, and could be a diagonal matrix if the tasks are completely unrelated. A first simple idea to learn the optimal matrix is to run an expert algorithm (e.g., Hedge) on a covering of the set of interaction matrices. However, it is too big a set for this approach to work. The way AdaTask learns the best interaction matrix is completely different.

Building on a novel regularizer, which can be seen as a regularized version of the Schatten 1-norm, we introduce an augmented potential, defined on both the task predictors and the interaction matrix. The regret bound we derive holds for any comparator (i.e., any choice of predictors and interaction matrix), and therefore holds in particular for the best interaction matrix in hindsight. We strongly believe that our generic approach, which consists in viewing a parameter we want to optimize as a comparator, can be used to solve problems beyond multitask OCO. Note that when the optimal matrix only depends on external information, such as the observed loss gradients, it might not be necessary to rely on such a general method, see e.g., AdaGrad (McMahan and Streeter, 2010; Duchi et al., 2011), or (Kuzborskij and Cesa-Bianchi, 2017) that uses directly the gradient outer product matrix to adapt to the smoothness of the regression function. The rest of the paper is organized as follows. In Section 2, we describe the multitask online learning setting. In Section 3, we introduce and analyze AdaTask, showing that it enjoys better regret guarantees than independent learning when the task activations are stochastic. In Section 4, we provide some details about its algorithmic implementation and present some experiments.

|       | Regret bound | a)       | b)       |
|-------|--------------|---------|---------|
| Cesa-Bianchi et al. (2021) | $\mathcal{O} \left( \sqrt{1 + (N - 1)\text{Var}(U)} \sqrt{T} \right)$ | $\mathcal{O}(\sqrt{T})$ | $\mathcal{O}(\sqrt{NT})$ |
| This paper | $\mathcal{O} \left( \|U\|_{S(1)} \sqrt{T/N} \right)$ | $\mathcal{O}(\sqrt{T})$ | $\mathcal{O}(\sqrt{T})$ |
2 Multitask Online Learning

Framework. We now formally introduce the multitask online learning framework studied in this paper. We consider a set of $N \in \mathbb{N}$ learning tasks, with decision space $\mathbb{R}^d$. At each time step $t = 1, 2, \ldots$ a task $i_t \in \{1, \ldots, N\}$ is activated, and the learner is asked to make a prediction $x_t \in \mathbb{R}^d$ for this task. The learner then incurs the loss $\ell_t(x_t)$, where $\ell_t : \mathbb{R}^d \to \mathbb{R}$ is some convex function, and observes $g_t \in \partial \ell_t(x_t)$, a subgradient of $\ell_t$ at point $x_t$. For any unknown (possibly adversarial) sequence $\ell_1, \ell_2, \ldots$ of convex losses, the goal of the learner is to minimize the multitask regret for any horizon $T$. This is defined as the sum of the regrets for each task

$$R_T = \sum_{t=1}^{T} \sum_{i_t = i} \ell_t(x_t) - \inf_{u \in \mathbb{R}^d} \sum_{t: i_t = i} \ell_t(u).$$

We introduce the following multitask notation. For $t \geq 1$, let $X_t \in \mathbb{R}^{N \times d}$ be the multitask predictor, such that $[X_t]_{i,:}$ is the prediction maintained by the learner at time step $t$ for task $i$, where $M_i$ denotes the $i$-th row of the matrix $M$. Similarly, for a set of comparators $u_1, \ldots, u_N$, let $U \in \mathbb{R}^{N \times d}$ be the multitask comparator storing the $u_i$ in rows. Finally, the multitask subgradient $G_t \in \mathbb{R}^{N \times d}$ is zero everywhere except at row $i_t$, where it is equal to $g_t$. Equipped with this notation, we have $R_T = \sup_{U \in \mathbb{R}^{N \times d}} R_T(U)$, where

$$R_T(U) := \sum_{t=1}^{T} \ell_t([X_t]_{i_t,:}) - \ell_t(U_{i_t,:}) \leq \sum_{t=1}^{T} \langle G_t, X_t - U \rangle.$$

The rightmost expression is the linearized regret, which bounds the regret from above using the convexity of $\ell$. Hence, we can restrict ourselves to linear losses defined over $\mathbb{R}^{N \times d}$ by $X \mapsto \langle G, X \rangle$.

A standard way to address multitask learning is to introduce communication between the tasks—see, e.g., Evgeniou et al. (2005). In online learning, this can be achieved using Follow-The-Regularized-Leader (FTRL) with a Mahalanobis potential $\psi_A(X) = \|X\|_A^2 := \text{Tr}(AX^T)$ for a fixed symmetric positive definite matrix $A \in \mathbb{S}^+_N$. (Cavallanti et al., 2010; Cesa-Bianchi et al., 2021). Indeed, FTRL with potential $\psi_A$ and constant learning rate $\eta > 0$ on linear losses writes

$$\forall t, \quad X_t = \arg\min_{X \in \mathbb{R}^{N \times d}} \text{Tr}(AX^T) + \eta \sum_{s=1}^{t-1} \langle G_s, X \rangle,$$

and is equivalent to (see Appendix A.1 for details)

$$\forall t, \quad [X_t]_{i,:} = \arg\min_{x \in \mathbb{R}^d} \|x\|_2^2 + \eta \sum_{s=1}^{t-1} \langle A_{i,s}^{-1} g_s, x \rangle.$$  \hspace{1cm} (2)

We see from (2) that at every time step $t$ all predictors $i$ (not only the active one $i_t$) get updated. Intuitively, through $g_t$ we learn about task $i_t$, and then share this feedback with the other tasks. Note, however, that the updates use scaled subgradients $A_{i_t}^{-1} g_t$. The scaling by $A_{i_t}^{-1}$ quantifies how much the learner believes tasks $i$ and $i_t$ are similar. If $i = i_t$ then only the active task $i_t$ is updated at time step $t$, and the algorithm reduces to running $N$ independent instances of FTRL. One can verify that $\psi_A$ is $2$-strongly convex with respect to $\| \cdot \|_A$, whose dual norm is $\| \cdot \|_{A^{-1}}$. Furthermore, note that for subgradients $g_t$ with Euclidean norm bounded by $L$, we have $\|G_t\|_{A^{-1}}^2 = A_{i_t,i_t}^{-1} \|g_t\|_2^2 \leq A_{i_t,i_t}^{-1} L^2$. Applying the standard FTRL analysis to (1), we thus obtain

$$\forall U \in \mathbb{R}^{N \times d}, \quad R_T (U) \leq \frac{\text{Tr}(AUU^T)}{\eta} + \frac{\eta L^2}{4} \sum_{i=1}^{T} A_{i,i_t}^{-1}. $$  \hspace{1cm} (3)

One is then encouraged to choose $A$ (and $\eta$) that make the right-hand side of (3) as small as possible. Cesa-Bianchi et al. (2021) show that choosing $A = (1 + N)I_N - I_N$ yields a regret that scales with $\sqrt{1 + \text{Var}(U)} \cdot (N - 1)$, where $\text{Var}(U) = \frac{1}{N^2} \sum_i \|U_i - U\|_2^2$, with $U = \frac{1}{N} \sum_j U_j$, is the task variance. This improves upon the $\sqrt{N}$ dependence achieved by $A = I_N$ (i.e., $N$ independent instances of FTRL) whenever $\text{Var}(U) \leq 1$, meaning when the tasks are similar enough. However, note that this choice for $A$ is invariant with respect to permutations of the tasks. As a consequence, it
cannot exploit general task structures, see Figure 1. A natural question is then whether there exists an algorithm that, without any prior knowledge on the task structure, has regret satisfying

\[
\forall U \in \mathbb{R}^{N \times d}, \quad R_T(U) \leq \inf_{\eta > 0, A \in \mathbb{S}^{N \times N}_+} \left\{ \frac{\text{Tr}(AUU^\top)}{\eta} + \frac{\eta L^2}{4} \sum_{t=1}^{T} A^{-1}_{i,i} \right\},
\]

where \( \mathbb{S}^{N \times N}_+ \) denotes the set of \( N \times N \) positive definite matrices. Note that (4) is an extremely strong bound: the regret scales in terms of the best possible matrix \( A \in \mathbb{S}^{N \times N}_+ \) with respect to the unknown comparator \( \bar{U} \). In Section 3, we show that if the task activations are stochastic, then the expected regret (w.r.t. the task activations) of AdaTask satisfies (4) up to logarithmic factors in \( T \) and \( \|U\|_{S(1)} \).

**Target bound.** In order to gain some intuition on the right-hand side of (4), note that in the absence of further assumptions, optimizing \( \eta \) yields the upper bound

\[
L \sqrt{\text{Tr}(AUU^\top) \sum_{t=1}^{T} A^{-1}_{i,i}} \leq L \sqrt{\text{Tr}(UU^\top) \sum_{t=1}^{T} A^{-1}_{i,i}} \leq L \sqrt{\text{Tr}(AUU^\top) \max_i A^{-1}_{ii} \sqrt{T}}.
\]

The above inequality is tight when the only active task is \( \text{argmax}_i A^{-1}_{ii} \). The upper bound (5) is, however, very difficult to optimize in \( A \) over the set of positive definite matrices. Simple closed form solutions can only be derived for particular families of matrices \( A \), e.g., parametric classes for which the optimization reduces to a scalar problem (Cavallanti et al., 2010). Using the looser upper bound \( L \sqrt{\text{Tr}(AUU^\top) \lambda_{\max}(A^{-1}) \sqrt{T}} \) yields a solvable yet unsatisfactory tradeoff in \( A \), as the best matrix in this case is \( I_N \) (independent FTRL), see Appendix A.1 for details. In order to obtain more informative bounds, we make the following assumption.

**Assumption 1 (STOCHASTIC ACTIVATIONS).** There exist \( \pi_1, \ldots, \pi_N > 0 \), with \( \sum_i \pi_i = 1 \), such that for all \( t \) and \( i \in \{1, \ldots, N\} \) it holds \( \mathbb{P}(i_t = i) = \pi_i \). We set \( \Pi = \text{diag}(\pi_1, \ldots, \pi_N) \in \mathbb{R}^{N \times N} \).

Note that Assumption 1 does not affect the \( g_t \), that may remain adversarial. We also highlight that considering stochastic activations is not a limitation with respect to the batch setting, where the numbers of observations per task are known in advance. To understand why stochastic activations help, note that choosing an interaction matrix \( A \) corresponds to creating communication edges between tasks, that is essential to propagate information. However, due to the term \( \text{Tr}(AUU^\top) \), these edges become a burden if they link tasks that are never activated. With stochastic activations, the learner can trade-off communication with activation by weighing each edge according to the activation probabilities of the tasks at its endpoints. Under Assumption 1, by taking the expectation on both sides of (3) and minimizing with respect to \( \eta \), we obtain the following upper bound on the expected regret with respect to \( U \)

\[
L \sqrt{\text{Tr}(AUU^\top) \text{Tr}(\Pi A^{-1}) \sqrt{T}}.
\]

When \( UU^\top \) is invertible, the above expression is minimized at \( A = \left( \Pi^{-1} UU^\top \right)^{-1/2} \), where it takes the value \( L \|U\|_{S(1)} \sqrt{T} \), see Appendix A.1 for details. For the simplicity of the exposition, we consider from now on uniform activations, i.e., \( \pi_i = 1/N \) for all \( i \in \{1, \ldots, N\} \). The generalization to arbitrary probabilities is deferred to Appendix A.8. The optimal choice becomes \( A^* = (UU^\top)^{-1/2} \), revealing that update (2) is optimized when the gradients are scaled using a root of the task covariance matrix \( UU^\top \). The final regret bound is

\[
\forall U \in \mathbb{R}^{N \times d}, \quad \mathbb{E}[R_T(U)] \leq L \|U\|_{S(1)} \sqrt{T/N}.
\]

By the Cauchy-Schwarz inequality, the right-hand side of (6) is always smaller than \( L \|U\|_{S(2)} \sqrt{T} \), which is the regret bound obtained with \( A = I_N \). The maximal improvement one can achieve is a \( \sqrt{N} \) factor, when all the tasks are equal. Interestingly, (6) provides a natural interpolation between important particular cases. Indeed, consider the case where the tasks are perfectly grouped into \( K \leq N \) independent clusters \( C_1, \ldots, C_K \) of equal sizes. In other words, \( U_i = e_j \) for all \( i \in C_j \), where \( e_j \in \mathbb{R}^d \) is the canonical base vector with non-zero entry \( j \). Then, the spectrum of \( UU^\top \) has exactly \( K \) non-zero eigenvalues, all equal to \( \sqrt{N/K} \), such that \( \|U\|_{S(1)} = \sqrt{N/K} \), and the right hand
We highlight that the regret bound (9) is only five times larger than the target bound (6) restricted to \( K = 1 \), i.e., all tasks are equal, we recover the single task regret bound. When \( K = N \), i.e., all tasks are independent, we recover a trivial upper bound. In the next section, we show that AdaTask satisfies (6) up to logarithmic factors in \( T \) and \( \|U\|_{S(1)} \). Note that (6) is an oracle bound, obtained by tuning \( A \) and \( \eta \) in FTRL with knowledge of the comparator \( U \), and assuming that \( U U^\top \) is invertible.

3 AdaTask

In this section, we introduce AdaTask and show that it satisfies the regret bound (6) up to logarithmic factors in \( T \) and \( \|U\|_{S(1)} \). To do that, we first design an algorithm with small regret on the Schatten 1-norm unit ball \( B_{S(1)} \subset \mathbb{R}^{N \times d} \). Using techniques from parameter-free OCO, we then extend these guarantees to all matrices in \( \mathbb{R}^{N \times d} \), at the cost of extra logarithmic factors. All missing proofs can be found in the Appendices. In what follows, \( \sigma_i(A) \) denotes the singular values of a matrix \( A \in \mathbb{R}^{N \times d} \).

Tight regret bound on \( B_{S(1)} \). Let

\[
\psi(X) := \text{Tr}\left((XX^\top + \lambda I_N)^{1/2}\right) = \sum_{i=1}^{N} \sqrt{\sigma_i^2(X) + \lambda},
\]

for some \( \lambda > 0 \) to be chosen later. The regularizer \( \psi \), which to the best of our knowledge is new, can be viewed as a strongly convex version of the Schatten 1-norm, see Figure 4 for a plot in the 1-dimensional case. In Theorem 1, we show that FTRL run with \( \psi \), which we refer to as Schatten-1 FTRL, has good regret guarantees uniformly over \( B_{S(1)} \). Our Schatten-1 FTRL update rule can be written as follows:

\[
\forall t, \quad X_t = \text{argmin}_{X \in B_{S(1)}} \psi(X) + \langle \eta_{t-1}H_{t-1}, X \rangle,
\]

where \( (\eta_t)_{t \geq 0} \) is a decreasing sequence of learning rates, and \( H_{t-1} = \sum_{s=1}^{t-1} G_s \). For completeness, the global online multitask learning interaction mechanism is also recalled in Algorithm 2.

**Theorem 1.** Suppose that the task activations satisfy Assumption 1 with \( \pi_i = 1/N \) for all \( i \), and that the loss functions \( \ell_t \) have subgradients with Euclidean norms bounded by \( L \). Then, Schatten-1 FTRL with \( \lambda = 1/N^2 \) and \( \eta_{t-1} = \sqrt{N}/(L \sqrt{T}) \) produces a sequence \((X_t)_{t \geq 1}\) such that

\[
\forall U \in B_{S(1)}, \quad \mathbb{E}[R_T(U)] \leq 5L \sqrt{T/N}.
\]

We highlight that the regret bound (9) is only five times larger than the target bound (6) restricted to \( B_{S(1)} \), and improves upon the \( L \sqrt{T} \) bound attained using independent FTRL by a factor \( \sqrt{N} \). Note that Theorem 1 cannot be proved via a standard analysis of Schatten-1 FTRL. Indeed, the fact that \( \psi = \lambda/(1 + \lambda)^{3/2} \)-strongly convex with respect to the Frobenius norm is not enough to derive the improved bound (9), see Appendix B.1 for technical details. The main reason is that the Frobenius norm is unable to exploit the randomization in the activation mechanism, and cannot therefore provide tight bounds in the stochastic case. Instead, (6) crucially relies on the use of the Mahalanobis norm \( \| \cdot \|_A \), that introduces the entries \( A_{i,i}^{-1} \) into the bound. When the activations \( i_t \) are stochastic, we have \( \mathbb{E}[A_{i_t,i_t}^{-1}] = \text{Tr}(A^{-1})/N \), leading to the \( \sqrt{N} \) improvement. The cornerstone of our analysis is that Mahalanobis norms can be recovered from \( \psi \) through the following variational expression

\[
\psi(X) = \frac{1}{\alpha} \inf_{A \in S_N^{++}} \frac{1}{2} \left( \text{Tr}\left(A(XX^\top + \lambda I_N)\right) + \alpha^2 \text{Tr}(A^{-1}) \right), \quad \forall \alpha > 0.
\]

Note that substituting (10) into (8), it is immediate to check that the update rule

\[
\forall t, \quad X_t, A_t = \text{argmin}_{X \in B_{S(1)}, A \in S_N^{++}} \phi(X, A) + \langle \eta'_{t-1}H_{t-1}, X \rangle,
\]

with \( \eta'_{t} = \alpha \eta_{t} \), generates the exact same sequence of iterates \((X_t)_{t \geq 1}\) as update (8). From now on, we thus focus on the analysis of (11). Interestingly, we do not require the joint strong convexity of \( \phi \), but only the strong convexity of \( \phi \) with respect to \( X \) (resp. \( A \)) when \( A \) (resp. \( X \)) is fixed.
Proposition 2. For any $A \in \mathbb{S}_N^+$, $\phi(\cdot, A)$ is 1-strongly convex with respect to $\| \cdot \|_A$. Furthermore, let $\delta > 0$ and $\mathcal{S}_\delta = \{ A \in \mathbb{S}_N^+ : A \prec \delta I_N \}$. For any $X \in \mathbb{R}^{N \times d}$, $\phi(X, \cdot)$ is $(\alpha^2 / \delta^3)$-strongly convex on the open convex set $\mathcal{S}_\delta$ with respect to the Frobenius norm.

Equipped with Proposition 2, we can now state a regret bound for the sequence generated by (11).

Theorem 3. Suppose that the task activations satisfy Assumption 1 with $\pi_t = 1/N$ for all $i$, and that the loss functions $\ell_t$ have subgradients with Euclidean norms bounded by $L$. Then, update (11) produces a sequence $(X_t, A_t)_{t \geq 1}$ such that for all $U \in \mathcal{B}_{S(1)}$ and $B \in \mathbb{S}_N^+$ we have

$$\mathbb{E}[R_T(U)] \leq \frac{\phi(U, B)}{\alpha \eta_{t-1}} + \left( \frac{L^2 (1 + \sqrt{\lambda} N)}{2N} + \frac{4 \alpha^2}{\lambda^{3/2}} \right) \sum_{i=1}^T \eta_{t-1}. \quad (12)$$

We now take a close look at the three terms in (12). The first one is the regularizer evaluated at the comparator. Note that, by definition of $\phi$, we actually have two comparators here: the predictors $U \in \mathbb{R}^{N \times d}$ and the matrix $B \in \mathbb{S}_N^+$. This feature will be key to derive (9) from (12). The second and the third terms derive from the strong convexity of $\phi$, with respect to $X$ and $A$ respectively. Note the role played by the non-zero regularization parameter $\lambda$, which prevents the last term from exploding. Bound (12) displays a trade-off between the three parameters $\alpha$, $\lambda$, and $\eta_{t-1}$. Solving this trade-off and choosing an appropriate matrix $B$ allows us to prove Theorem 1.

Proof of Theorem 1. As discussed above, the sequences $(X_t)_{t \geq 1}$ produced by updates (8) and (11) are the same. We can thus start from bound (12) and look for the best choice of $B$, $\alpha$, $\lambda$, and $\eta_{t-1}$. Indeed, since bound (12) is valid for any $B \in \mathbb{S}_N^+$, it holds true in particular for $B = \alpha(UU^\top + \lambda I_N)^{-1/2}$. We then obtain $\phi(U, B) = \alpha \sum_{i=1}^N \sqrt{\sigma_i^2(U)} + \lambda \leq \alpha(1 + \sqrt{\lambda} N)$. Substituting this upper bound into (12), with $\eta_{t-1} = \gamma / \sqrt{t}$, gives

$$\forall U \in \mathcal{B}_{S(1)}, \quad \mathbb{E}[R_T(U)] \leq \left( \frac{1 + \sqrt{\lambda} N}{\gamma} + \frac{L^2 (1 + \sqrt{\lambda} N)}{N} + \frac{8 \alpha^2 \gamma}{\lambda^{3/2}} \right) \sqrt{T}. \quad (13)$$

Setting $\lambda = 1/N^2$, $\gamma = \sqrt{N} / L$, and $\alpha = L / \sqrt{N} N^2$, yields the desired bound. \qed

The main novelty of our analysis is the technique we use to learn the best positive definite interaction matrix $A$ when the regret is measured against an arbitrary comparator $U$, see (4). In order to achieve this, we define an augmented comparator $(U, B) \in \mathbb{R}^{N \times d} \times \mathbb{S}_N^+$, and introduce a potential $\phi$ over $X$ and $A$. The strong convexity of $\phi$ then allows us to derive the regret bound (12) for any interaction matrix, and thus in particular for the one minimizing the right-hand side of (4). We strongly believe that our general approach, which consists in viewing a parameter we want to optimize as a comparator, can be used to solve problems beyond multitask OCO. Note however that constructing the augmented potential $\phi$ might be challenging in general. In our case, with $\alpha$ set to 1 for simplicity, $\phi(X, A)$ is the sum of two terms: the Mahalanobis norm $\text{Tr}(AXX^\top)$, and the quantity $\lambda \text{Tr}(A) + \text{Tr}(A^\top)$. Minimizing $\phi(X_t, A)$ with respect to $A$ then gives $A_t = (XX_t^\top + \lambda I_N)^{-1/2}$, which is a reasonable proxy for $A^* = (UU^\top)^{-1/2}$. Finally, note that considering a potential with extra arguments creates new descent directions, possibly with better curvature, explaining the improvement when the augmented potential is used. For more technical details, we provide a precise quantitative comparison in Appendix B.1. To conclude this paragraph, we provide a lower bound that adapts standard arguments from OCO to show that Schatten-1 FTRL is optimal up to constants.

Theorem 4. Suppose that the task activations satisfy Assumption 1 with $\pi_t = 1/N$ for all $i$, and consider linear losses such that $\ell_t(x) = \langle g_t, x \rangle = \langle G_t, X \rangle$. Then, there exists a sequence of vectors $g_1, \ldots, g_T \in \mathbb{R}^d$, with $\|g_t\|_2 \leq L$ for all $t$, such that for any sequence $X_1, \ldots, X_T$ we have

$$\sup_{U \in \mathcal{B}_{S(1)}} \mathbb{E}[R_T(U)] \geq L \sqrt{T / (2N)}. \quad (14)$$

Extension to $\mathbb{R}^{N \times d}$. Our next challenge consists in extending Theorem 1 to any $U \in \mathbb{R}^{N \times d}$. This problem, known as parameter-free OCO, has been intensively studied in the past few years. In particular, Cutkosky and Orabona (2018) propose an elegant way to build parameter-free strategies by combining two base algorithms: a 1-dimensional online algorithm $A_{1d}$ is used to learn the norm of
Algorithm 1 AdaTask

\begin{algorithm}
\begin{algorithmic}
\caption{AdaTask}
\begin{itemize}
\item \textbf{input}: Number of tasks $N$, parameter $\lambda > 0$, learning rates $(\eta_i)_{i \geq 0}$, Lipschitz constant $L$
\item \textbf{init}: $H = 0_{R^N \times d}$
\item \textbf{for} $t = 1$ to $T$ \textbf{do}
\begin{itemize}
\item Compute $\bar{X}_t = \arg\min_{X \in \mathcal{B}_S(1)} \text{Tr} \left( (XX^T + \lambda I_N)^{1/2} \right) + \langle \eta_{t-1} H, X \rangle$ \hspace{1cm} \triangleright \text{Schatten-1 FTRL}
\item Compute $y_t = -\frac{1}{T} \sum_{i=1}^{t-1} s_i \left( 1 - \sum_{i=1}^{t-1} s_i y_i \right)$ \hspace{1cm} \triangleright \text{KT-OCO}
\item Compute $X_t = y_t \bar{X}_t$
\item Receive $i_t$ drawn uniformly at random from $\{1, \ldots, N\}$
\item Predict $x_{i_t} = [X_t]_{i_t}$
\item Pay $\ell_t(x_{i_t})$ and receive $g_t \in \partial \ell_t(x_{i_t})$
\item Update $H$: $[H]_{i_t} \leftarrow [H]_{i_t} + g_t$ \hspace{1cm} \triangleright \text{gradient update for the Schatten-1 FTRL}
\item Set $s_t = \frac{\sqrt{N}}{T} \left( g_{i_t}, \bar{X}_t \right)$ \hspace{1cm} \triangleright \text{gradient update for the KT-OCO}
\end{itemize}
\end{itemize}
\end{algorithmic}
\end{algorithm}

Theorem 5. Suppose that the task activations satisfy Assumption 1 with $\pi_i = 1/N$ for all $i$, and that the loss functions $\ell_i$ have subgradients with Euclidean norms bounded by $L$. Then, AdaTask (Algorithm 1) with $\lambda = 1/N^2$ and $\eta_{t-1} = \sqrt{N}/(L \sqrt{t})$ produces a sequence $(X_t)_{t \geq 1}$ such that

$$\forall U \in \mathbb{R}^{N \times d}, \quad E[R_T(U)] \leq L \|U\|_{S(1)} \sqrt{\frac{T}{N}} \left( 5 + 2 \sqrt{\log \left( 1 + C_0 \|U\|_{S(1)} T \right)} \right) + \frac{L}{\sqrt{N}}. \quad (14)$$

Related works. Learning multiple tasks jointly with their structure was already addressed by Ciliberto et al. (2015), albeit in a batch framework, with i.i.d. losses and a vector-valued Reproducing Kernel Hilbert Space as model space. Although their barrier method—see (Ciliberto et al., 2015, Theorem 3.3)—reminds a batch version of Schatten-1 FTRL, it is based on alternate minimization in $X$ and $A$, whereas our analysis requires simultaneous minimization. More importantly, they do not provide any theoretical guarantee on the excess risk attained by their method. On the contrary, via a multitask online-to-batch conversion (see Theorem 6, which we state with arbitrary activation probabilities but in $\mathcal{B}_S(1)$ for simplicity), we can show that our algorithm applied to i.i.d. losses enjoys a small excess risk. Note that Theorem 6 provides simpler proofs to some of the multitask bounds established by Pontil and Maurer (2013).

Theorem 6 (MULTITASK ONLINE-TO-BATCH). Suppose that each task $i$ is defined by a distribution $\mu_i$ over some set $Z_i$ and a $L$-Lipschitz convex loss function $\ell^{(i)} : \mathbb{R}^d \times Z_i \to \mathbb{R}$. For any $X \in \mathbb{R}^{N \times d}$, the multitask risk is $R_{MT}(X) = \sum_{i=1}^{N} \pi_i \mathcal{R}_i(X_{i_t})$, where $\mathcal{R}_i(x) = E_{z \sim \mu_i}[\ell^{(i)}(x, z^{(i)})]$ is the risk associated to task $i$. Suppose that we are given a sequence of tasks and examples $(i_1, z_1), \ldots, (i_T, z_T)$, where $i_t$ is drawn from a distribution $\pi_1, \ldots, \pi_N$ over $\{1, \ldots, N\}$, and $z_t$ is drawn from $\mu_{i_t}$. We note $z_t = (i_t, z_t)$, and $X_1, \ldots, X_T$ the predictors generated by AdaTask from the sequence $(z_t)_{t \geq 1}$. Let $\bar{X}_T = (1/T) \sum_{t=1}^{T} X_{i_t}$, we have

$$E_{z_1, \ldots, z_T} [R_{MT}(\bar{X}_T)] \leq \inf_{U \in \mathcal{B}_S(1)} R_{MT}(U) + \frac{5L}{\sqrt{NT}}.$$
Moreover, if each $\ell \in [0,1]$, then with probability at least $1 - \delta$ (over the random draw of $\tilde{z}_i$), we have that

$$E_{i_1, \ldots, i_T} [R_{MT}(\tilde{X}_T)] \leq \inf_{U \in B_{s(1)}} R_{MT}(U) + \frac{5L}{\sqrt{NT}} + \sqrt{\frac{8}{T} \log \frac{2}{\delta}}.$$  

Another approach to multitask learning is to assume that all the tasks share a small set of features. Learning then consists in finding both the predictors and some linear transformations of the inputs, under a sparsity-inducing penalization (Argyriou et al., 2008). This approach is somehow dual to ours, as the interactions are enforced on the features rather than on the outputs (the tasks). Expressed in our formalism, this amounts to using the potential defined on $\mathbb{R}^{N \times d} \times S_{d}^{++}$ by

$$\phi'(X, A) = \text{Tr}(AX^\top X) + \text{Tr}(A) + \text{Tr}(A^{-1}),$$

where we have omitted the scaling factors in front of the trace terms for simplicity. The potential $\phi'$ differs from (7) due to the term $\text{Tr}(AX^\top X)$, instead of $\text{Tr}(AXX^\top)$. In Appendix B.2, we show that this potential cannot yield the improved regret bound (9). This happens because the Mahalanobis norm $\text{Tr}(AX^\top X)$ is not adapted to the sparsity of the matrix gradients $G_t$. A possibility to make the potential (15) work is to assume stochastic feature activations, instead of stochastic task activations, so that the gradient matrices $G_t$ become column-sparse (instead of row-sparse), see Appendix B.2 for technical details. As far as we know, this is the first time one identifies a regime in which the multitask approach proposed in (Argyriou et al., 2008) is provably better than independent learning. However, the assumptions needed appear to be a less justifiable than ours in practice.

It is also worth discussing the parallel between AdaTask and AdaGrad (McMahan and Streeter, 2010; Duchi et al., 2011). Indeed, yet another way to write the Schatten-1 FTRL update is

$$\forall t, \quad Y_t = \arg\min_{Y \in B_{s(1)}} \text{Tr}(M_{t-1}YY^\top) + \langle \eta_{t-1}H_{t-1}, Y \rangle,$$

where $M_{t-1} = (X_tX_t + \lambda I_N)^{-1/2}$, with $X_t$ computed from (8). Note that $M_{t-1}$ only depends (through $X_t$) on the past gradients $G_1, \ldots, G_{t-1}$, hence the time indexing. It can be checked (see Proposition 9 in Appendix B.1) that $Y_t = X_t$ for all $t \geq 1$. On the other hand, AdaGrad’s update rule on feasible set $F$ (McMahan and Streeter, 2010, Equation (5)) writes

$$\forall t, \quad x_t = \arg\min_{x \in F} \frac{1}{2} \sum_{s=1}^{t-1} (x - x_s)^\top Q_s (x - x_s) + \langle \sum_{s=1}^{t-1} g_s, x \rangle,$$

for some matrices $Q_s \in \mathbb{R}^{d \times d}$ to be defined later. From the perspective of update (16), AdaTask and AdaGrad both use time-indexed Mahalanobis regularizers to achieve a similar goal: establishing a regret bound expressed as an infimum over a set of matrices. The resemblance, however, ends here. First, AdaGrad is a Follow-The-Proximally-Regularized-Leader algorithm, as the regularizers are centered at the successive iterates $x_s$, while our regularizer is centered at the origin. Second, AdaGrad is a single-task algorithm, while we exploit the task similarity in a multitask and matrix-valued setting. As a consequence, the matrices used in the Mahalanobis norms are fundamentally different. AdaGrad uses diagonal matrices $Q_s$, whose entries depend on the coordinates of the gradients seen so far, see (McMahan and Streeter, 2010, Corollaries 1 and 2). AdaTask uses $M_{t-1} = (X_tX_t + \lambda I_N)^{-1/2}$, the inverse square root of the regularized covariance matrix of the predictor $X_t$. The subsequent analyses of the induced dependencies are therefore completely different. We also highlight that competitive ratios for AdaGrad are computed after taking the supremum of the Bregman divergence over the comparator set, see (McMahan and Streeter, 2010, Equations (10) and (11)), while we almost reach the performance attained by a comparator-dependent tuning of $A$.

We note also that (Argyriou et al., 2007, Proposition 4.2) provides a variational formulation for Schatten $p$-norms that is similar to (10). Recall that Schatten $p$-norms regularizers have been previously studied in the literature. Duchi et al. (2010, Section 7.3) use $p$-norms for Mirror Descent, but selects $p$ to be close to 1, instead of regularizing $\| \cdot \|_{2(1)}$ using $\lambda$ as in (7). Note however that their framework is completely different from ours (in particular it is not multitask). Kakade et al. (2012) also investigate Schatten-$p$ norm regularizers, but again for $p > 1$ and only with a focus on Schatten unit-balls, while we provide a parameter-free approach. Their bounds are quite generic, and cannot leverage the stochasticity of the activations since they miss the variational formulation (10). More generally, one may refer to (Zhang and Yang, 2021) for a recent survey on multitask learning. In the
Figure 2: On the 2 clusters dataset, Schatten-1 FTRL fully exploits the task structure, including the negative correlations, and improves upon Independent FTRL. On the contrary, Variance FTRL shares information between tasks that have a big variance and suffers a large regret, see Figure 2(a). On the Lenk dataset, which is known to contain (very) similar tasks, AdaVar is extremely efficient as compared to independent learning, see Figure 2(b). AdaTask is worse than AdaVar at first, but outperforms it in the long run. Indeed, AdaVar only requires to learn the task variance, while AdaTask needs to identify the best possible interaction matrix. This estimation problem is harder and takes more time. When the optimal matrix is identified however, AdaTask is bound to outperform AdaVar.

On SARCOS, AdaTask outperforms both AdaVar and Independent, without any delay (Figure 2(c)).

online setting, it is worth mentioning the work by Abernethy et al. (2007), that is limited to prediction with expert advice though, and that of Dekel et al. (2007); Murugesan et al. (2016), although they do not consider asynchronous activations like we do. For meta-learning applications, see Alquier et al., 2017; Finn et al., 2019; Balcan et al., 2019; Denevi et al., 2019), and (Zhang and Yeung, 2010; Pentina and Lampert, 2017; Shui et al., 2019) for different ways of learning task similarities. Finally, note the recent works by Herbster et al. (2021) for an analysis of multitask multi-armed bandits, and by Boursier et al. (2022) for multitask learning guarantees with trace norm regularization when the number of samples per task is small.

4 Algorithm and Experiments

Computation of $X_t$. From an algorithmic point of view, an interesting feature of AdaTask is that computing update (8) reduces to a $N$-dimensional problem with a strongly convex objective.

Proposition 7. Let $H_{t-1} = U_{t-1} D_{t-1} V_{t-1}^\top$, with $D_{t-1} = \text{diag}(d_1, \ldots, d_N)$, be the Singular Value Decomposition (SVD) of $H_{t-1}$. Then, we have $X_t = U_{t-1} \Sigma^* V_{t-1}^\top$, where $\Sigma^* = \text{diag}(\sigma_1^*, \ldots, \sigma_N^*)$ and $\sigma^* = (\sigma_1^*, \ldots, \sigma_N^*) \in \mathbb{R}^N$ is the solution to

$$
\min_{\|\sigma\|_1 \leq 1} \sum_{i=1}^{N} \sqrt{\sigma_i^2 + \lambda} + \eta_{t-1} \sum_{i=1}^{N} d_i \sigma_i.
$$

(17)

Note that the objective function of problem (17) is $\lambda/(1 + \lambda)^{3/2}$-strongly convex, and can thus be solved at rate $O(1/k)$, for example via projected subgradient descent (Beck, 2017, Theorem 8.31. A limitation of the above approach is that the SVD must be computed at each time step. Recall, however, that this allows to sequentially learn the best possible interaction matrix, making this computational overhead a reasonable compromise. We also highlight that $H_t - H_{t-1} = G_t$ is of rank 1, so that one can use the SVD of $H_{t-1}$ to compute faster that of $H_t$ (Bunch et al., 1978; Gandhi and Rajgor, 2017).

Synthetic data experiments. We reproduced the task structure of Figure 1, b). We generated $N = 100$ task vectors $\{u_i\}_{i \leq 100} \in \mathbb{R}^{10}$, spread across 2 clusters with opposite directions. The matrix $U$ is rescaled such that $\|U\|_{S(1)} = 1$ (the $u_i$ then have an individual squared norm around 0.05). The inputs $x_t$ are generated uniformly on the sphere, and the labels are $y_t = \langle u_i, x_t \rangle + \epsilon_t$, where $\epsilon_t \sim U[-0.01, 0.01]$. We run the algorithms with the square loss for $T = 10000$, i.e., 100 points per task in expectation. We compared Schatten-1 FTRL against Independent FTRL, that does not use the
task structure, and Variance FTRL, i.e., Mahalanobis FTRL with $A = (1 + N)I_N - I_N^T$, following (Cesa-Bianchi et al., 2021). All parameters are set to their theoretical values. Figure 2(a) shows the results for 5 repetitions. Note that additional clustered task structures are studied in Appendix C.

**Real data experiments.** We next tested the parameter-free extensions of the above three algorithms, that we refer to as AdaTask, Independent, and AdaVar, on two standard multitask datasets: Lenk (Lenk et al., 1996) and SARCOS (Li et al., 2015). In Lenk, the input points are descriptions of 16 computers using 14 binary features, and the labels are ratings on the scale \{1, \ldots, 10\} made by 18 individuals (the tasks), for a total of $T = 2880$ (we did several passes on the dataset to better showcase the asymptotic behavior). The goal is to predict the likelihood of an individual to buy a new computer given its features. SARCOS is related to an inverse dynamic problem on a robot arm. It consists of 44,484 inputs of dimension 21 (7 positions, 7 velocities, 7 accelerations) that we should map to 7 torques (the tasks), for a total of $T = 311,388$. On both datasets we used the square loss and theoretical values to tune the algorithms (inputs and outputs were normalized, and $L$ computed accordingly). Results are reported in Figures 2(b) and 2(c) for 5 and 3 repetitions respectively.

**Acknowledgments and Disclosure of Funding**

L. R. and A. D. V. acknowledge the financial support of the European Research Council (grant SLING 819789), the AFOSR project FA9550-18-1-7009 (European Office of Aerospace Research and Development) and the EU H2020-MSCA-RISE project NoMADS - DLV-777826.

**References**

Abernethy, J., Bartlett, P., and Rakhlin, A. (2007). Multitask learning with expert advice. In *International Conference on Computational Learning Theory*, pages 484–498.

Alquier, P., Mai, T. T., and Pontil, M. (2017). Regret bounds for lifelong learning. In *Artificial Intelligence and Statistics*, pages 261–269. PMLR.

Argyriou, A., Evgeniou, T., and Pontil, M. (2008). Convex multi-task feature learning. *Machine learning*, 73(3):243–272.

Argyriou, A., Pontil, M., Ying, Y., and Micchelli, C. (2007). A spectral regularization framework for multi-task structure learning. *Advances in neural information processing systems*, 20.

Balcan, M.-F., Khodak, M., and Talwalkar, A. (2019). Provable guarantees for gradient-based meta-learning. In *International Conference on Machine Learning*, pages 424–433. PMLR.

Beck, A. (2017). *First-order methods in optimization*. SIAM.

Boursier, E., Konobeev, M., and Flammarion, N. (2022). Trace norm regularization for multi-task learning with scarce data. *arXiv preprint arXiv:2202.06742*.

Bunch, J. R., Nielsen, C. P., and Sorensen, D. C. (1978). Rank-one modification of the symmetric eigenproblem. *Numerische Mathematik*, 31(1):31–48.

Caruana, R. (1997). Multitask learning. *Machine Learning*, 28(1):41–75.

Cavallanti, G., Cesa-Bianchi, N., and Gentile, C. (2010). Linear algorithms for online multitask classification. *The Journal of Machine Learning Research*, 11:2901–2934.

Cesa-Bianchi, N., Laforgue, P., Paudice, A., and Pontil, M. (2021). Multitask online mirror descent. *arXiv preprint arXiv:2106.02393*.

Ciliberto, C., Mroueh, Y., Poggio, T., and Rosasco, L. (2015). Convex learning of multiple tasks and their structure. In *International Conference on Machine Learning*, pages 1548–1557. PMLR.

Cutkosky, A. and Orabona, F. (2018). Black-box reductions for parameter-free online learning in banach spaces. In *Conference On Learning Theory*, pages 1493–1529. PMLR.
Dekel, O., Long, P. M., and Singer, Y. (2007). Online learning of multiple tasks with a shared loss. *Journal of Machine Learning Research*, 8(10).

Denevi, G., Stamos, D., Ciliberto, C., and Pontil, M. (2019). Online-within-online meta-learning. In *Advances in Neural Information Processing Systems* 32, volume 32, pages 1–11.

Duchi, J., Hazan, E., and Singer, Y. (2011). Adaptive subgradient methods for online learning and stochastic optimization. *Journal of machine learning research*, 12(7).

Duchi, J. C., Shalev-Shwartz, S., Singer, Y., and Tewari, A. (2010). Composite objective mirror descent. In *COLT*, volume 10, pages 14–26. Citeseer.

Evgeniou, T., Micchelli, C. A., and Pontil, M. (2005). Learning multiple tasks with kernel methods. *Journal of Machine Learning Research*, 6(4).

Finn, C., Rajeswaran, A., Kakade, S., and Levine, S. (2019). Online meta-learning. In *International Conference on Machine Learning*, pages 1920–1930. PMLR.

Gandhi, R. and Rajgor, A. (2017). Updating singular value decomposition for rank one matrix perturbation. *arXiv preprint arXiv:1707.08369*.

Hazan, E. (2016). Introduction to online convex optimization. *Foundations and Trends in Optimization*, 2(3-4):157–325.

Herbster, M., Pasteris, S., Vitale, F., and Pontil, M. (2021). A gang of adversarial bandits. *Advances in Neural Information Processing Systems*, 34.

Juditsky, A. and Nemirovski, A. S. (2008). Large deviations of vector-valued martingales in 2-smooth normed spaces. *arXiv preprint arXiv:0809.0813*.

Kakade, S. M., Shalev-Shwartz, S., and Tewari, A. (2012). Regularization techniques for learning with matrices. *The Journal of Machine Learning Research*, 13(1):1865–1890.

Kuzborskij, I. and Cesa-Bianchi, N. (2017). Nonparametric online regression while learning the metric. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pages 667–676.

Lenk, P. J., DeSarbo, W. S., Green, P. E., and Young, M. R. (1996). Hierarchical bayes conjoint analysis: Recovery of partworth heterogeneity from reduced experimental designs. *Marketing Science*, 15(2):173–191.

Li, Y., Tian, X., Liu, T., and Tao, D. (2015). Multi-task model and feature joint learning. In *Twenty-Fourth International Joint Conference on Artificial Intelligence*.

McMahan, H. B. and Streeter, M. (2010). Adaptive bound optimization for online convex optimization. *arXiv preprint arXiv:1002.4908*.

Murugesan, K., Liu, H., Carbonell, J., and Yang, Y. (2016). Adaptive smoothed online multi-task learning. In *Advances in Neural Information Processing Systems*, pages 4296–4304.

Orabona, F. (2019). A modern introduction to online learning. *arXiv preprint arXiv:1912.13213*.

Ostrowski, A. (1952). Sur quelques applications des fonctions convexes et concaves au sens de I. Schur. *J. Math. Pures Appl*, 31(9):253–292.

Pentina, A. and Lampert, C. H. (2017). Multi-task learning with labeled and unlabeled tasks. In *International Conference on Machine Learning*, pages 2807–2816. PMLR.

Pontil, M. and Maurer, A. (2013). Excess risk bounds for multitask learning with trace norm regularization. In *Conference on Learning Theory*, pages 55–76. PMLR.

Saha, A., Rai, P., Daumé, H., Venkatasubramanian, S., et al. (2011). Online learning of multiple tasks and their relationships. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, pages 643–651. JMLR Workshop and Conference Proceedings.
Shui, C., Abbasi, M., Robitaille, L.-É., Wang, B., and Gagné, C. (2019). A principled approach for learning task similarity in multitask learning. In Proceedings of the 28th International Joint Conference on Artificial Intelligence (IJCAI-19), pages 3446–3452.

Zhang, Y. and Yang, Q. (2021). A survey on multi-task learning. IEEE Transactions on Knowledge and Data Engineering.

Zhang, Y. and Yeung, D. Y. (2010). A convex formulation for learning task relationships in multi-task learning. In Proceedings of the 26th Conference on Uncertainty in Artificial Intelligence, page 733.
A Technical Proofs

In this section we gather all the technical proofs that have been omitted in the core text.

A.1 Omitted technical details

Multitask FTRL. Recall that multitask FTRL with potential \( \psi(X) = \text{Tr}(AXX^\top) \) writes

\[
\forall t, \quad X_t = \arg\min_{X \in \mathbb{R}^{N \times d}} \text{Tr}(AXX^\top) + \eta \sum_{s=1}^{t-1} \langle G_s, X \rangle.
\]

By differentiating with respect to \( X \) we obtain

\[
X_t = -\frac{\eta}{2} A^{-1} \sum_{s=1}^{t-1} G_s,
\]

that by the sparsity of the matrix gradients \( G_s \) can be rewritten

\[
\forall t, \forall i \in \{1, \ldots, N\}, \quad [X_t]_i = -\frac{\eta}{2} \sum_{s=1}^{t-1} A_{ii}^{-1} g_s.
\]

It is then immediate to check that computing

\[
\forall t, \forall i \in \{1, \ldots, N\}, \quad [X_t]_i = \arg\min_{x \in \mathbb{R}^d} \|x\|^2_2 + \eta \sum_{s=1}^{t-1} \langle A_{ii}^{-1} g_s, x \rangle
\]

is equivalent.

**On the minimization of** \( \text{Tr}(A U U^\top) \cdot \lambda_{\text{max}}(A^{-1}) \). We have the following sequence of equivalent problems

\[
\min_{A \in \mathbb{S}_N^+} \text{Tr}(A U U^\top) \cdot \lambda_{\text{max}}(A^{-1})
\]

\[
\min_{A \in \mathbb{S}_N^+} \text{Tr}(A^{-1} U U^\top) \cdot \lambda_{\text{max}}(A)
\]

\[
\min_{A \in \mathbb{S}_N^+} \text{Tr}(A^{-1} U U^\top) \quad \text{s.t.} \quad \lambda_{\text{max}}(A) = 1
\]

\[
\min_{A \in \mathbb{S}_N^+} \text{Tr}(A^{-1} U U^\top) \quad \text{s.t.} \quad \lambda_{\text{max}}(A) \leq 1
\]

\[
\min_{d_1, \ldots, d_N} \sum_{i=1}^N \frac{\sigma_i^2(U)}{d_i} \quad \text{s.t.} \quad d_i \leq 1 \quad \forall i,
\]

where the last equivalence comes from the fact that the optimal \( A \) admits the eigenvalue decomposition \( A = P D P^\top \), where \( D = \text{diag}(d_1, \ldots, d_N) \) and \( U = \Sigma Q^\top \) is the SVD of \( U \), see e.g., the proof of Theorem 7.18 in Beck (2017). From the last problem, it is immediate to check that the solution is given by \( d_i = 1 \) for all \( i \), i.e., \( A = I_N \).

**On the minimization of** \( \text{Tr}(A U U^\top) \cdot \text{Tr}(\Pi A^{-1}) \). Similarly, the following problems are equivalent

\[
\min_{A \in \mathbb{S}_N^+} \text{Tr}(A U U^\top) \cdot \text{Tr}(\Pi A^{-1})
\]

\[
\min_{B \in \mathbb{S}_N^+} \text{Tr}(B \Pi U U^\top) \cdot \text{Tr}(B^{-1})
\]

\[
\min_{B \in \mathbb{S}_N^+} \text{Tr}(B \Pi U U^\top) \quad \text{s.t.} \quad \text{Tr}(B^{-1}) = 1
\]

\[
\min_{B \in \mathbb{S}_N^+} \text{Tr}(B \Pi U U^\top) \quad \text{s.t.} \quad \text{Tr}(B^{-1}) \leq 1,
\]

where we have used the substitution \( B = A \Pi^{-1} \). Now, the last problem is convex, and a standard Lagrangian analysis yields that the optimal \( B \) is \( (\Pi U U^\top)^{-1/2} \), or equivalently that the optimal \( A \) is \( (\Pi^{-1} U U^\top)^{-1/2} \).
A.2 Proof of Proposition 2

Recall that for all $X \in B_{S(1)}$ and $A \in S_{N+}^+$ we have

$$\phi(X, A) = \frac{1}{2} \left( \text{Tr}(A X X^\top + \lambda I_N) \right) + \alpha^2 \text{Tr}(A^{-1}) \right).$$

**Proposition 2.** For any $A \in S_{N+}^+$, $\phi(\cdot, A)$ is 1-strongly convex with respect to $\| \cdot \|_A$. Furthermore, let $\delta > 0$ and $S_\delta = \{ A \in S_{N+}^+ : A < \delta I_N \}$. For any $X \in \mathbb{R}^{N \times d}$, $\phi(X, \cdot)$ is $(\alpha^2/\delta^3)$-strongly convex on the open convex set $S_\delta$ with respect to the Frobenius norm.

**Proof.** The first claim is immediate to check, e.g., by differentiating twice $\phi(\cdot, A)$. We recall that for all $X \in \mathbb{R}^{N \times d}$ and any $A \in S_{N+}^+$, we use the notation $\|X\|_A^2 = \text{Tr}(A X X^\top)$. For the second claim, let us first recall that a sufficient condition for a twice differentiable function $F : V \subset \mathbb{R}^{N \times N} \to \mathbb{R}$ to be $\mu$-strongly convex with respect to $\| \cdot \|_{F \delta}$ is that for any $A, H \in V$ holds

$$\langle D^2 F(A)[H], H \rangle \geq \mu \|H\|_{F \delta}^2,$$

where $D^2 F(A) : \mathbb{R}^{N \times N} \to \mathbb{R}^{N \times N}$ denotes the second order differential operator of $F$ at point $A$. Note that $\phi(X, A)$ is the sum of the linear term (in $A$) $\frac{1}{2} \text{Tr}(A X X^\top + \lambda I_N)$ and the convex term $\alpha^2 \text{Tr}(A^{-1})$. It is thus enough to show that the latter is $(\alpha^2/\delta^3)$-strongly convex on $S_\delta$. To do so, observe that $\text{Tr}(A^{-1}) = \sum_{i=1}^N 1/\lambda_i(A)$, such that we can use Proposition 3.1 from Juditsky and Nemirovski (2008) with $\Delta = (0, \delta)$ and $f(x) = 1/x$. Let $F(A) = \text{Tr}(A^{-1})$. By (26) in Juditsky and Nemirovski (2008), we have that for any $A \in S_\delta$ and any symmetric matrix $H \in S_N$ it holds

$$\langle D^2 F(A)[H], H \rangle = \sum_{s,t} \Gamma_{s,t}[f] \tilde{H}_{st}^2,$$

where $\tilde{H} = U^\top H U$, with $U$ being the matrix whose columns are the eigenvectors of $A$, and

$$\Gamma_{s,t}[f] = \left\{ \begin{array}{ll} f'(\lambda_s(A)) - f'(\lambda_t(A)) \lambda_s(A) - \lambda_t(A), & \lambda_s(A) \neq \lambda_t(A) \\ f''(\lambda_s(A)), & \lambda_s(A) = \lambda_t(A) \end{array} \right.$$

Note here that we can avoid resorting to condition (11) from Juditsky and Nemirovski (2008) and bound the derivatives directly. Indeed, we have $f''(x) = 2/x^3$ and $\max_s \lambda_s(A) \leq \delta$, such that $f''(\lambda_s(A)) \geq 2/\delta^3$. Similarly, by the mean value theorem, there exists $z \in [\lambda_s(A), \lambda_t(A)]$ such that

$$\frac{f'(\lambda_s(A)) - f'(\lambda_t(A))}{\lambda_s(A) - \lambda_t(A)} = f''(z) \geq 2/\delta^3.$$

Substituting these values into (18), we obtain

$$\langle D^2 F(A)[H], H \rangle \geq \frac{2}{\delta^3} \sum_{s,t} \tilde{H}_{st}^2 = \frac{2}{\delta^3} \|\tilde{H}\|_{F \delta}^2 = \frac{2}{\delta^3} \|H\|_{F \delta}^2,$$

where the last equality follows from the definition of $\tilde{H}$. Multiplying (19) by $\alpha^2/2$ concludes the proof. \hfill \square

A.3 Proof of Theorem 3

Recall that we use the update rule

$$\forall t, \ X_t, A_t = \text{argmin}_{X \in B_{S(1)}, A \in S_{N+}^+} \phi(X, A) + \langle \eta_{t-1} H_{t-1}, X \rangle,$$

where $\eta_t = \alpha \eta_t$, $(\eta_t)_{t \geq 0}$ is a decreasing sequence of learning rates, and $H_{t-1} = \sum_{s=1}^{t-1} G_s$.

**Theorem 3.** Suppose that the task activations satisfy Assumption 1 with $\pi_i = 1/N$ for all $i$, and that the loss functions $\ell_i$ have subgradients with Euclidean norms bounded by $L$. Then, update (11) produces a sequence $(X_t, A_t)_{t \geq 1}$ such that for all $U \in B_{S(1)}$ and $B \in S_{N+}^+$ we have

$$\mathbb{E}[R_T(U)] \leq \frac{\phi(U, B)}{\alpha \eta_{T-1}} + \left( \frac{L^2 (1 + \sqrt{X})}{2N} + \frac{4\alpha^2}{\lambda^{3/2}} \right) \sum_{t=1}^T \eta_{t-1}. \quad (12)$$
Proof. Let $\phi_t = \phi/\eta_{t-1}$. Applying Lemma 7.1 from Orabona (2019) to the sequence of linear loss functions $X \mapsto \langle G_t, X \rangle$, we obtain that for all $U \in B_{S(1)}$ and $B \in S_N^{++}$ we have

$$R_T(U) = \sum_{t=1}^{T} \ell_t([X_t]_{i,t}) - \ell_t(U_{i,t})$$

$$\leq \sum_{t=1}^{T} \langle G_t, X_t - U \rangle$$

$$\leq \phi_T(U, B) + \sum_{t=1}^{T} \left| F_t(X_t, A_t) - F_{t+1}(X_{t+1}, A_{t+1}) + \langle G_t, X_t \rangle \right|,$$

where $F_t(X, A) = \phi_t(X, A) + \langle H_{t-1}, X \rangle$. Furthermore, we have

$$F_t(X_t, A_t) - F_{t+1}(X_{t+1}, A_{t+1}) + \langle G_t, X_t \rangle$$

$$= \phi_t(X_t, A_t) + \langle H_t, X_t \rangle - \phi_{t+1}(X_{t+1}, A_{t+1}) - \langle H_t, X_{t+1} \rangle$$

$$= \phi_t(X_t, A_t) + \langle H_t, X_t \rangle - \phi_t(X_{t+1}, A_t) - \langle H_t, X_{t+1} \rangle$$

$$+ \phi_t(X_{t+1}, A_t) - \phi_{t+1}(X_{t+1}, A_t) + \phi_{t+1}(X_{t+1}, A_t) - \phi_{t+1}(X_{t+1}, A_{t+1})$$

$$\leq \phi_t(X_t, A_t) + \langle H_t, X_t \rangle - \phi_t(X^*_t, A_t) - \langle H_t, X^*_t \rangle$$

$$+ \phi_{t+1}(X_{t+1}, A_t) - \phi_{t+1}(X_{t+1}, A_{t+1}),$$

where $X^*_t = \arg\min_{X \in B_{S(1)}} \phi_t(X, A_t) + \langle H_t, X \rangle$. Note that Equation (20) (respectively (21)) accounts for the learning of $X$ (respectively $A$) when $A$ (respectively $X$) is fixed. We can then individually bound these terms using the strong convexity of $\phi_t$ established in Proposition 2. By Corollary 7.7 in Orabona (2019), we have

$$\phi_t(X_t, A_t) + \langle H_t, X_t \rangle - \phi_t(X^*_t, A_t) - \langle H_t, X^*_t \rangle \leq \frac{\eta_{t-1}}{2} \| G_t \|^2_{A_t^{-1}} - \frac{\alpha \eta_{t-1}}{2} \| G_t \|^2_{A_t^{-1}},$$

where $G'_t \in \partial \left( \phi_t(\cdot, A_t) + \langle H_t, \cdot \rangle + \chi_{B_{S(1)}} \right)(X_t)$, with $\chi_{S}$ the characteristic function of a set $S$ such that $\chi_{S}(x) = 0$ if $x \in S$ and $\chi_{S}(x) = +\infty$ otherwise. However, note that we have $0 \in \partial \left( \phi_t(\cdot, A_t) + \langle H_t, \cdot \rangle + \chi_{B_{S(1)}} \right)(X_t)$, so that by additivity of subdifferentials we can take $G'_t = G_t$. Recalling that $G_t$ has only one non-zero row (the row $i_t$), we have

$$E \left[ \| G_t \|^2_{A_t^{-1}} \right] = E \left[ \| A_t^{-1} \|_{1,i_t} \| g_t \|^2_{2} \right] = \frac{Tr(A_t^{-1})}{N} \| g_t \|^2_{2} \leq \frac{Tr(A_t^{-1})}{N} L^2.$$

Furthermore, it holds

$$Tr(A_t^{-1}) = \frac{1}{\alpha} Tr \left( (X_t^t X_t^T + \lambda I_N)^{1/2} \right)$$

$$= \frac{1}{\alpha} \sum_{i=1}^{N} \sqrt{\sigma_i^2(X_t)} + \lambda \leq \frac{X_t \|_{S(1)} + \sqrt{\lambda} N}{\alpha} \leq 1 + \sqrt{\lambda} N.$$

Overall, we obtain

$$E \left[ \phi_t(X_t, A_t) + \langle H_t, X_t \rangle - \phi_t(X^*_t, A_t) - \langle H_t, X^*_t \rangle \right] \leq \frac{L^2(1 + \sqrt{\lambda} N)}{2N} \eta_{t-1}.$$

Regarding (21), note that we have $A_t \in S_{2\alpha/\sqrt{\lambda}}$ for all $t$, such that Proposition 2 combined with Corollary 7.7 from Orabona (2019) yields

$$\phi_{t+1}(X_{t+1}, A_t) - \phi_{t+1}(X_{t+1}, A_{t+1}) \leq \frac{\alpha \eta_{t}}{\lambda^{3/2}} \| X_{t+1}^T X_{t+1}^T + \lambda I_N - \alpha^2 A_t^{-2} \|_{Fr}$$

$$= \frac{\alpha^2 \eta_{t}}{\lambda^{3/2}} \| X_{t+1}^T X_{t+1}^T - X_t^T X_t^T \|_{Fr}^2$$

$$\leq \frac{4 \alpha^2 \eta_{t}}{\lambda^{3/2}} \sup \| X_t \|_{S(4)} \leq \frac{4 \alpha^2}{\lambda^{3/2}} \eta_{t-1}.$$
Collecting all the inequalities, we finally obtain that for all $U \in \mathcal{B}_{S(1)}$ and $B \in \mathbb{S}^N_+$ it holds

$$
\mathbb{E} [R_T(U)] \leq \frac{\phi(U, B)}{\alpha \eta_{T-1}} + \left( \frac{L^2 (1 + \sqrt{\lambda} N)}{2N} + \frac{4\alpha^2}{\lambda^{3/2}} \right) \sum_{t=1}^T \eta_{t-1}.
$$

\[\square\]

### A.4 Proof of Theorem 4

**Theorem 4.** Suppose that the task activations satisfy Assumption 1 with $\pi_i = 1/N$ for all $i$, and consider linear losses such that $\ell_t(x) = \langle g_t, x \rangle = \langle G_t, X \rangle$. Then, there exists a sequence of vectors $g_1, \ldots, g_T \in \mathbb{R}^d$, with $\|g_t\|_2 \leq L$ for all $t$, such that for any sequence $X_1, \ldots, X_T$ we have

$$
\sup_{U \in \mathcal{B}_{S(1)}} \mathbb{E} [R_T(U)] \geq L \sqrt{T/(2N)}.
$$

**Proof.** Let $X_1, \ldots, X_T$ be any sequence and $\bar{R}_T = \sup_{U \in \mathcal{B}_{S(1)}} \mathbb{E} [R_T(U)]$. Let $g_t = \epsilon_t Le_1$, where $e_1 = (1, 0, \ldots, 0) \in \mathbb{R}^d$ and $\epsilon_t \in \{-1, 1\}$. We show that there exists a sequence of $\epsilon_t$ such that $R_T$ is lower bounded by $L \sqrt{T/2N}$. To do so, we use that $\sup_{\epsilon \in \{-1, 1\}} R_T \geq \mathbb{E}_{P \sim \mathcal{P}} [\bar{R}_T]$ for any probability distribution $P$ over $\{-1, 1\}$. In particular, we chose $P$ to be such that $\mathbb{P} (\epsilon_t = -1) = 1/2$. For simplicity, from now on we drop the $P$ notation in the expectation with respect $\epsilon = (\epsilon_1, \ldots, \epsilon_T)$. We also introduce $E_1 \in \mathbb{R}^{N \times d}$ such that $[E_1]_{i,:} = \epsilon_1 / \sqrt{N}$ for all $i \leq N$. It is immediate to check that $E_1 \in \mathcal{B}_{S(1)}$ and that $\|E_1\|_{Fr} = 1$. We have

$$
\sup_{\epsilon_1, \ldots, \epsilon_T \in \{-1, 1\}} \bar{R}_T \geq \mathbb{E}_{\epsilon} \left[ \sup_{U \in \mathcal{B}_{S(1)}} \mathbb{E}_{\epsilon_1, \ldots, \epsilon_T} \left[ \sum_{t=1}^T \langle g_t, [X_t]_{i_t} \rangle - \langle g_t, U_{i_t} \rangle \right] \right]
$$

$$
= \frac{1}{N} \mathbb{E}_{\epsilon} \left[ \sup_{U \in \mathcal{B}_{S(1)}} \sum_{t=1}^T \sum_{i=1}^N \langle g_t, [X_t]_{i_t} \rangle - \langle g_t, U_{i_t} \rangle \right]
$$

$$
= \frac{L}{\sqrt{N}} \mathbb{E}_{\epsilon} \left[ \sum_{t=1}^T \epsilon_t \langle E_1, X_t \rangle - \inf_{U \in \mathcal{B}_{S(1)}} \sum_{t=1}^T \epsilon_t \langle E_1, U \rangle \right]
$$

$$
\geq \frac{L}{\sqrt{N}} \mathbb{E}_{\epsilon} \left[ \sum_{t=1}^T \epsilon_t \langle E_1, U \rangle \right]
$$

$$
\geq \frac{L \|E_1\|_{Fr}^2}{\sqrt{N}} \mathbb{E}_{\epsilon} \left[ \sum_{t=1}^T \epsilon_t \right]
$$

$$
\geq L \sqrt{\frac{T}{2N}},
$$

where the last inequality follows from Khintchine inequality. \[\square\]

### A.5 Proof of Theorem 5

**Theorem 5.** Suppose that the task activations satisfy Assumption 1 with $\pi_i = 1/N$ for all $i$, and that the loss functions $\ell_t$ have subgradients with Euclidean norms bounded by $L$. Then, AdaTask (Algorithm 1) with $\lambda = 1/N^2$ and $\eta_{T-1} = \sqrt{N} / (L \sqrt{T})$ produces a sequence $(X_t)_{t \geq 1}$ such that

$$
\forall U \in \mathbb{R}^{N \times d}, \quad \mathbb{E} [R_T(U)] \leq L \|U\|_{S(1)} \sqrt{\frac{1}{N} \left( 5 + 2 \sqrt{\log (1 + C_0 \|U\|_{S(1)}T)} \right)} + \frac{L}{\sqrt{N}}.
$$
Proof. Recall that all expectations are taken here with respect to the task activations. We have
\[
\mathbb{E}[R_T(U)]
\]
\[
= \mathbb{E} \left[ \sum_{t=1}^{T} \ell_t([X_t]_{i_t}) - \ell_t(U_{i_t}) \right]
\]
\[
\leq \mathbb{E} \left[ \sum_{t=1}^{T} \langle G_t, X_t - U \rangle \right]
\]
\[
= \mathbb{E} \left[ \sum_{t=1}^{T} \langle G_t, y_t \tilde{X}_t - U \rangle \right]
\]
\[
= \mathbb{E} \left[ \sum_{t=1}^{T} \langle G_t, y_t \tilde{X}_t - \|U\|_{S(1)} \tilde{X}_t \rangle \right] + \mathbb{E} \left[ \sum_{t=1}^{T} \langle G_t, \|U\|_{S(1)} \tilde{X}_t - U \rangle \right]
\]
\[
= \frac{L}{\sqrt{N}} \mathbb{E} \left[ \sum_{t=1}^{T} \sqrt[N]{N} \langle G_t \tilde{X}_t \rangle (y_t - \|U\|_{S(1)}) \right] + \|U\|_{S(1)} \mathbb{E} \left[ \sum_{t=1}^{T} \langle G_t, \tilde{X}_t - \frac{U}{\|U\|_{S(1)}} \rangle \right].
\]

(23)

Quantity \(A\) is the expected linear regret of the KT-OCO algorithm. By the linearity of the expectation, it is equal to the expected linear regret of the KT-OCO algorithm provided with expected gradients
\[
A = \mathbb{E} \left[ \sum_{t=1}^{T} \sqrt[N]{N} \mathbb{E}_{i_t} \left[ \langle G_t, \tilde{X}_t \rangle \mid i_1, \ldots, i_{t-1} \right] (y_t - \|U\|_{S(1)}) \right].
\]

In order to apply the regret bound (13), we have to check that the expected gradients have absolute value smaller than 1. Let \(M_t = (\tilde{X}_t \tilde{X}_t^T + \delta I_N)^{-1/2}\) for some \(\delta > 0\), we have
\[
\left| \langle G_t, \tilde{X}_t \rangle \right| \leq \|G_t\|_{M_t^{-1}} \|\tilde{X}_t\|_{M_t}^2
\]
\[
= [M_t^{-1}]_{i_t i_t} \|g_t\|^2_2 \sum_{i=1}^{N} \frac{\sigma_i^2(\tilde{X}_t)}{\sigma_i^2(\tilde{X}_t) + \delta}
\]
\[
\leq \frac{L^2}{\delta} [M_t^{-1}]_{i_t i_t} \sum_{i=1}^{N} \sqrt[2]{\sigma_i^2(\tilde{X}_t) + \delta}
\]
\[
\leq L^2 (1 + \sqrt{\delta N}) [M_t^{-1}]_{i_t i_t}.
\]

Consequently, for any \(\delta > 0\) we have
\[
\left| \frac{\sqrt[N]{N}}{L} \mathbb{E}_{i_t} \left[ \langle G_t, \tilde{X}_t \rangle \right] \right| \leq N(1 + \sqrt{\delta N}) \mathbb{E}_{i_t} \left[ \sqrt[2]{[M_t^{-1}]_{i_t i_t}}^2 \right]
\]
\[
\leq N(1 + \sqrt{\delta N}) \mathbb{E}_{i_t} \left[ [M_t^{-1}]_{i_t i_t} \right]
\]
\[
= (1 + \sqrt{\delta N}) \text{Tr}(M_t^{-1})
\]
\[
= (1 + \sqrt{\delta N}) \sum_{i=1}^{N} \sqrt{\sigma_i^2(\tilde{X}_t) + \delta}
\]
\[
\leq (1 + \sqrt{\delta N})^2.
\]

The latter holding true for any \(\delta > 0\), we have that the expected gradients have absolute vaule smaller than 1. Using the regret bound (13) for KT-OCO with gradients in \([-1, 1]\) (Orabona, 2019, Chapter 9) we obtain
\[
A \leq 2\|U\|_{S(1)} \sqrt{T \log (1 + C_0\|U\|_{S(1)}T)} + 1,
\]

(24)
Moreover, if each
we have
where
Theorem 6
(Multitask Online-to-Batch).
Suppose that each task \(i\) is defined by a distribution \(\mu_i\) over some set \(\mathcal{Z}\), and a \(L\)-Lipschitz convex loss function \(\ell(i); \mathbb{R}^d \times \mathcal{Z} \rightarrow \mathbb{R}\). For any \(X \in \mathbb{R}^{N \times d}\), the multitask risk is \(R_{\text{MT}}(X) = \sum_{i=1}^{N} \pi_i \mathcal{R}_i(X_i)\), where \(\mathcal{R}_i(x) = \mathbb{E}_{z^i \sim \mu_i}[\ell(i)(x, z^i)]\) is the risk associated to task \(i\). Suppose that we are given a sequence of tasks and examples \((i_1, z_1), \ldots, (i_T, z_T)\), where \(i_t\) is drawn from a distribution \(\pi_1, \ldots, \pi_N\) over \(\{1, \ldots, N\}\), and \(z_t\) is drawn from \(\mu_{i_t}\). We note \(\tilde{z}_t = (i_t, z_t)\). Using the above inequality for the multitask risk is \(\mathcal{R}_{\text{MT}}\left(\tilde{z}_t\right)\), it is easy to check that
\[\mathcal{R}_{\text{MT}}(\tilde{X}_T) = (1/T) \sum_{t=1}^{T} \mathcal{R}_{\text{MT}}(X_t, z_t)\]
To bound this, we note
\[\mathcal{R}_{\text{MT}}(\tilde{X}_T) \leq \inf_{U \in \mathcal{B}_{S(1)}} \mathcal{R}_{\text{MT}}(U) + \frac{5L}{\sqrt{NT}} + \sqrt{\frac{8T \log 2}{\delta}}\]
Applying (Orabona, 2019, Theorem 3.1), we obtain that for all \(U \in \mathcal{B}_{S(1)}\) it holds
\[\mathbb{E}_{\tilde{z}_1, \ldots, \tilde{z}_T}[\mathcal{R}_{\text{MT}}(\tilde{X}_T)] \leq \mathcal{R}_{\text{MT}}(U) + \frac{\mathbb{E}_{\tilde{z}_1, \ldots, \tilde{z}_T}[\sum_{t=1}^{T} \ell_{\text{MT}}(X_t, \tilde{z}_t) - \ell_{\text{MT}}(U, \tilde{z}_t)]}{T}\]
The proof of the first claim is concluded by observing that
\[\sum_{t=1}^{T} \ell_{\text{MT}}(X_t, \tilde{z}_t) - \ell_{\text{MT}}(U, \tilde{z}_t) = \sum_{t=1}^{T} \ell(i_t)(X_t, \tilde{z}_t) - \ell(i_t)(U, \tilde{z}_t) = R_T(U)\]
that can be bounded using Theorem 1.
For the second claim, (Orabona, 2019, Theorem 3.13) and the convexity of \(\mathcal{R}_{\text{MT}}\) yield that for any \(U \in \mathcal{B}_{S(1)}\) and any \(\delta\) we have that with probability at least \(1 - \delta\) it holds
\[\mathcal{R}_{\text{MT}}(\tilde{X}_T) \leq \mathcal{R}_{\text{MT}}(U) + \frac{R_T(U)}{T} + 2 \sqrt{\frac{2 \log 2}{\delta}}\]
Using the above inequality for the \(U\) that minimizes \(\mathcal{R}_{\text{MT}}\) over \(\mathcal{B}_{S(1)}\), and taking both sides the expectation with respect to the \(i_t\) (in order to use Theorem 1) concludes the proof. \(\square\)
A.7 Proof of Proposition 7

Proposition 7. Let \( H_{t-1} = U_{t-1}D_{t-1}V_{t-1}^T \), with \( D_{t-1} = \text{diag}(d_1, \ldots, d_N) \), be the Singular Value Decomposition (SVD) of \( H_{t-1} \). Then, we have \( X_t = U_{t-1} \Sigma^* V_{t-1}^T \), where \( \Sigma^* = \text{diag}(\sigma_1^*, \ldots, \sigma_N^*) \) and \( \sigma^* = (\sigma_1^*, \ldots, \sigma_N^*) \in \mathbb{R}^N \) is the solution to

\[
\min_{\|\sigma\|_1 \leq 1} \sum_{i=1}^N \sqrt{\sigma_i^2 + \lambda} + \eta_{t-1} \sum_{i=1}^N d_i \sigma_i.
\]  

(17)

Proof. For simplicity, we drop the time indexing in the notation for the following proof. Let \( H = UDV^T \) be the SVD of \( H \), and \( F : \mathbb{R}^{N \times d} \rightarrow \mathbb{R} \) such that

\[
F(X) = \eta(H, X) + \text{Tr}((XX^T + \lambda I_N)^{1/2})
\]

\[
= \langle \eta D, U^T XV \rangle + \text{Tr}((U^T XV)(U^T XV)^T + \lambda I_N)^{1/2}.
\]

Therefore, we have \( X^* = \arg\min_{X \in B_{S(1)}} F(X) = UY^* V^T \) with

\[
Y^* = \arg\min_{Y \in U^T B_{S(1)} V} \langle \eta D, Y \rangle + \text{Tr}((YY^T + \lambda I_N)^{1/2})
\]

\[
= \arg\min_{Y \in B_{S(1)}} \langle \eta D, Y \rangle + \text{Tr}((YY^T + \lambda I_N)^{1/2}) \quad (26)
\]

Now, let us show that \( Y^* \) is diagonal. We use the same method as in the proof of (Beck, 2017, Theorem 7.29). Let \( \Sigma_1 \in \mathbb{R}^{N \times N} \), \( \Sigma_2 \in \mathbb{R}^{d \times d} \) be diagonal matrices with all entries equal to 1 except the \( i \)-th one, equal to \( -1 \). Note that these matrices are orthogonal. Let \( Y_i^* = \Sigma_1 Y_i^* \Sigma_2^T \), we have

\[
G(Y_i^*) = \langle \eta D, Y_i^* \rangle + \text{Tr}((Y_i^* Y_i^* + \lambda I_N)^{1/2})
\]

\[
= \langle \eta \Sigma_1 D \Sigma_2^2, Y_i^* \rangle + \text{Tr}((Y_i^* Y_i^* + \lambda I_N)^{1/2})
\]

\[
= \langle \eta D, Y_i^* \rangle + \text{Tr}((Y_i^* Y_i^* + \lambda I_N)^{1/2})
\]

\[
= G(Y_i^*).
\]

So \( Y_i^* \) is also a solution to problem (26) (note that \( Y_i^* \in B_{S(1)} \)). By the uniqueness of the solution – recall that the objective function is strongly convex – we can conclude that \( Y^* \) is diagonal, i.e., that \( X^* = U \Sigma^* V^T \). Substituting this decomposition into \( F \), we have that \( \sigma^* \in \mathbb{R}^N \), such that \( \Sigma^* = \text{diag}(\sigma^*) \), is the solution to

\[
\min_{\|\sigma\|_1 \leq 1} \sum_{i=1}^N \sqrt{\sigma_i^2 + \lambda} + \eta \sum_{i=1}^N d_i \sigma_i,
\]

where the \( d_i \) are such that \( D = \text{diag}(d_i) \).

A.8 Extension to arbitrary activation probabilities

In this section, we derive the regret guarantees for AdaTask with generic activation probabilities \( \pi_1, \ldots, \pi_N \). For completeness, we first recall the stochastic activations assumption.

Assumption 1 (Stochastic activations). There exist \( \pi_1, \ldots, \pi_N > 0 \), with \( \sum_i \pi_i = 1 \), such that for all \( t \) and \( i \in \{1, \ldots, N\} \) it holds \( P\{i_t = i\} = \pi_i \). We set \( \Pi = \text{diag}(\pi_1, \ldots, \pi_N) \in \mathbb{R}^{N \times N} \).

Recall also that in this general framework the target regret bound is

\[
\forall U \in \mathbb{R}^{N \times d}, \quad \mathbb{E}[R_T(U)] \leq L \|\Pi^{1/2} U\|_{S(1)} \sqrt{T}.
\]  

(27)

Note that this bound is smaller than \( L \|U\|_{S(2)} \sqrt{T} \), attained with Euclidean FTRL. Indeed, applying (Ostrowski, 1952) and Cauchy-Schwarz inequality, we have

\[
\|\Pi^{1/2} U\|_{S(1)} = \sum_{i=1}^N \sigma_i(\Pi^{1/2} U) \leq \sum_{i=1}^N \sigma_i(\Pi^{1/2}) \sigma_i(U) \leq \sqrt{\sum_{i=1}^N \sigma_i^2(\Pi^{1/2}) \sum_{i=1}^N \sigma_i^2(U)} = \|U\|_{S(2)}.
\]

AdaTask achieves the target regret bound (27) up to logarithmic factor in \( \|\Pi^{1/2} U\|_{S(1)} \) and \( T \).
Theorem 8. Suppose that the task activations satisfy Assumption 1, and that the loss functions $\ell_t$ have subgradients with Euclidean norms bounded by $L$. Then, AdaTask (Algorithm 4) with $\lambda = 1/N^2$ and $\eta_{t-1} = 1/(L\sqrt{T})$ produces a sequence $(X_t)_{t \geq 1}$ such that

$$\forall U \in \mathbb{R}^{N \times d}, \quad \mathbb{E}[R_T(U)] \leq L \|\Pi^{1/2}U\|_{S(1)} \sqrt{T} \left( 5 + 2\log \left( 1 + C_0 \|\Pi^{1/2}U\|_{S(1)}T \right) \right) + L.$$  

Proof. The proof is analog to that for uniform task activations. First, we derive a regret bound on $B_{\Pi} = \{ U \in \mathbb{R}^{N \times d}; \|\Pi^{1/2}U\|_{S(1)} \leq 1\}$. We then extend the results to $\mathbb{R}^{N \times d}$. Note that by substituting $\Pi = I_N/N$ into $B_{\Pi}$ we obtain the Schatten 1-norm ball of radius $\sqrt{N}$, while we focused on the unit ball in the core text. This explains why we do not recover exactly the proof of Theorem 5 by substituting $\Pi = I_N/N$ into the following computations and algorithms. Up to this $\sqrt{N}$ rescaling, both analyses are equivalent though.

Bound on $B_{\Pi}$. Let

$$\psi_{\Pi}(X) := \text{Tr} \left( (\Pi^{1/2}XX^\top \Pi^{1/2} + \lambda I_N)^{1/2} \right) = \sum_{i=1}^N \sqrt{\sigma_i^2(\Pi^{1/2}X) + \lambda}.$$  

We show that the sequence

$$\forall t, \quad X_t = \arg\min_{X \in B_{\Pi}} \psi_{\Pi}(X) + \langle \eta_{t-1} H_{t-1}, X \rangle,$$

where $(\eta_t)_{t \geq 0}$ is a decreasing sequence of learning rates, and $H_{t-1} = \sum_{s=1}^{t-1} G_s$, enjoys a small regret on $B_{\Pi}$. To do so, we consider again the augmented regularizer

$$\phi_{\Pi}(X, A) = \frac{1}{2} \left( \text{Tr} \left( A(\Pi^{1/2}XX^\top \Pi^{1/2} + \lambda I_N) \right) + \alpha^2 \text{Tr}(A^{-1}) \right)$$

and the update rule

$$\forall t, \quad X_t, A_t = \arg\min_{X \in B_{\Pi}, A \in S_{+}^d} \phi_{\Pi}(X, A) + \langle \eta_{t-1} H_{t-1}, X \rangle,$$

where $\eta_t' = \alpha \eta_t$. Similarly to the case with uniform activations, the latter update generates the same sequence $(X_t)_{t \geq 1}$ as the one with the compact regularizer $\psi_{\Pi}$. By an analysis similar to that of Proposition 2, we can show that $\phi_{\Pi}(\cdot, A)$ is 1-strongly convex with respect to $\|\cdot\|_{\Pi^{1/2}A \Pi^{1/2}}$, and that $\phi_{\Pi}(X, \cdot)$ is $(\alpha^2/L^3)$-strongly convex on $S_\delta$ with respect to the Frobenius norm. Using the same analysis as in Theorem 3, we have

$$\phi_{\Pi,t}(X_t, A_t) + \langle H_t, X_t \rangle - \phi_{\Pi,t}(X_t^*, A_t) - \langle H_t, X_t^* \rangle \leq \frac{\alpha \eta_t}{2} \|G_t\|_{(\Pi^{1/2}A_{\Pi}^{1/2})^{-1}}^2,$$

with

$$\mathbb{E} \left[ \|G_t\|_{(\Pi^{1/2}A_{\Pi}^{1/2})^{-1}}^2 \right] = \mathbb{E} \left[ \|\Pi^{1/2}A_{\Pi}^{-1/2} \Pi^{1/2} - 1\|_{i,t}^2 \right] = \text{Tr}(A_t^{-1}) \|g_t\|_2^2 \leq \text{Tr}(A_t^{-1}) L^2$$

and

$$\text{Tr}(A_t^{-1}) = \frac{1}{\alpha} \text{Tr} \left( (\Pi^{1/2}XX^\top \Pi^{1/2} + \lambda I_N)^{1/2} \right) = \frac{1}{\alpha} \sum_{i=1}^N \sqrt{\sigma_i^2(\Pi^{1/2}X) + \lambda} \leq \frac{1 + \sqrt{\lambda N}}{\alpha},$$

where we have used that $A_t = \alpha (\Pi^{1/2}XX^\top \Pi^{1/2} + \lambda I_N)^{-1/2} \in S_{2\alpha/\sqrt{\lambda}}$ for all $t$. On the other hand, we have

$$\phi_{\Pi,t+1}(X_{t+1}, A_t) - \phi_{\Pi,t+1}(X_{t+1}, A_{t+1}) \leq \frac{\alpha \eta_t'}{\lambda^{3/2}} \left\| \Pi^{1/2}XX^\top \Pi^{1/2} + \lambda I_N - A_t^{-1} \right\|_{F,4}^2$$

$$= \frac{\alpha^2 \eta_t}{\lambda^{3/2}} \left\| \Pi^{1/2}XX^\top \Pi^{1/2} - \Pi^{1/2}XX^\top \Pi^{1/2} \right\|_{F,4}^2$$

$$\leq \frac{4\alpha^2 \eta_t}{\lambda^{3/2}} \sup_t \left\| \Pi^{1/2}XX^\top \right\|_{S(4)}^4$$

$$\leq \frac{4\alpha^2 \eta_t}{\lambda^{3/2}} \eta_{t-1}.$$

20
Gathering all inequalities, we obtain that for all $U \in B_{11}$ and $B \in S_N^+$ it holds
\[
E[R_T(U)] \leq \frac{\phi_{11}(U, B)}{\alpha \eta_{T-1}} + \left( \frac{L^2(1 + \sqrt{N})}{2} + \frac{4\alpha^2}{\lambda^{3/2}} \right) \sum_{t=1}^{T} \eta_{t-1}.
\]
With $B = \alpha(\Pi^{1/2}UU^T \Pi^{1/2} + \lambda I_N)^{-1/2}$, so that $\phi_{11}(U, B) = \alpha \text{Tr}((\Pi^{1/2}UU^T \Pi^{1/2} + \lambda I_N)^{1/2}) \leq \alpha(1 + \sqrt{N})$, and $\eta_{t-1} = \gamma / \sqrt{t}$, we obtain
\[
E[R_T(U)] \leq \left( \frac{1 + \sqrt{N}}{\gamma} + \gamma L^2(1 + \sqrt{N}) + \frac{8\gamma \alpha^2}{\lambda^{3/2}} \right) \sqrt{T}.
\]
Choosing $\lambda = 1/N^2$, $\gamma = 1/L$ and $\alpha = L / \sqrt{8N^3/2}$ finally yields
\[
\forall U \in B_{11}, \quad E[R_T(U)] \leq 5L \sqrt{T}.
\]
**Extension to $\mathbb{R}^{N \times d}$.** We use a similar decomposition as in the uniform case. Let $X_t$ generated by Algorithm 4, we have
\[
E[R_T(U)] \leq E \left[ \sum_{t=1}^{T} \langle G_t, X_t - U \rangle \right]
= E \left[ \sum_{t=1}^{T} \langle G_t, y_t \tilde{X}_t - U \rangle \right]
= E \left[ \sum_{t=1}^{T} \langle G_t, y_t \tilde{X}_t - \| \Pi^{1/2}U \|_{S(1)} \tilde{X}_t \rangle \right] + E \left[ \sum_{t=1}^{T} \langle G_t, \| \Pi^{1/2}U \|_{S(1)} \tilde{X}_t - U \rangle \right]
= L \sum_{t=1}^{T} \left( \frac{\langle G_t, \tilde{X}_t \rangle}{L} (y_t - \| \Pi^{1/2}U \|_{S(1)}) \right) + \| \Pi^{1/2}U \|_{S(1)} \sum_{t=1}^{T} \left( \frac{\langle G_t, \tilde{X}_t \rangle - U}{\| \Pi^{1/2}U \|_{S(1)}} \right)
\]
\[
(28)
\]
\[
(29)
\]
We have now to check that $E[\langle G_t, \tilde{X}_t \rangle] \leq L$. Let $M_t = \Pi^{1/2}(\Pi^{1/2} \tilde{X}_t \tilde{X}_t^T \Pi^{1/2} + \delta I_N)^{-1/2} \Pi^{1/2}$ for some $\delta > 0$, we have
\[
\left| \langle G_t, \tilde{X}_t \rangle \right|^2 \leq \| G_t \|_{M_t^{-1}}^2 \| \tilde{X}_t \|_{M_t}^2
= [M_t^{-1}]_{1i_1} \| g_t \|_{M_t}^2 \sum_{i=1}^{N} \frac{\sigma_{i}^2(\Pi^{1/2} \tilde{X}_t)}{\sqrt{\sigma_{i}^2(\Pi^{1/2} \tilde{X}_t) + \delta}}
\leq L^2 [M_t^{-1}]_{1i_1} \sum_{i=1}^{N} \frac{\sigma_{i}^2(\Pi^{1/2} \tilde{X}_t)}{\sigma_{i}^2(\Pi^{1/2} \tilde{X}_t) + \delta}
\leq L^2 (1 + \sqrt{\delta N}) [M_t^{-1}]_{1i_1},
\]
such that
\[
E[\langle G_t, \tilde{X}_t \rangle] \leq L \sqrt{(1 + \sqrt{\delta N}) E_{i_1} [M_t^{-1}]_{1i_1}}
= L \sqrt{(1 + \sqrt{\delta N}) \text{Tr}(\Pi M_t^{-1})}
\]
\[
(28)
\]
\[
(29)
\]
On the other hand, (29) can be bounded using the results on $B_{11}$. Gathering all inequalities yields the desired result.
B Additional technical discussions

In this section are collected two technical discussions. The first one regards the different ways of writing (and analyzing) Schatten-1 FTRL. The second deals with the dual approach to multitask learning considered in Argyriou et al. (2008). In particular we adapt the AdaTask analysis to show under which assumptions such an approach is provably better than independent learning.

B.1 Three shades of Schatten-1 FTRL

As already partially discussed in the core text, there are several ways of writing (and analyzing) Schatten-1 FTRL. Indeed, for all \( t \geq 1 \), let

\[
X_t = \arg\min_{X \in \mathcal{B}_{S(1)}} \text{Tr}((XX^\top + \lambda I_N)^{1/2}) + \langle \eta_{t-1} H_{t-1}, X \rangle, \quad (30)
\]

\[
Y_t = \arg\min_{Y \in \mathcal{B}_{S(1)}} \frac{1}{2} \text{Tr}(M_{t-1} YY^\top) + \langle \eta_{t-1} H_{t-1}, Y \rangle, \quad (31)
\]

\[
Z_t, A_t = \arg\min_{z \in \mathcal{B}_{S(1)}, \, A \in S_N^+} \frac{1}{2} \left( \text{Tr}(A(ZZ^\top + \lambda I_N)) + \text{Tr}(A^{-1}) \right) + \langle \eta_{t-1} H_{t-1}, Z \rangle, \quad (32)
\]

where \( M_{t-1} = (X_tX_t^\top + \lambda I_N)^{-1/2} \). Note that \( M_{t-1} \) only depends, through \( X_t \), on the past gradients \( G_1, \ldots, G_t \), hence the \( t-1 \) times indexing. The above three update rules actually generate the same sequence of iterates.

**Proposition 9.** For all \( t \geq 1 \), we have \( X_t = Y_t = Z_t \), and \( A_t = M_{t-1} \).

**Proof.** Let

\[
\psi(X) = \text{Tr}((XX^\top + \lambda I_N)^{1/2})
\]

and

\[
\phi(X, A) = \frac{1}{2} \left( \text{Tr}(A(XX^\top + \lambda I_N)) + \text{Tr}(A^{-1}) \right).
\]

We have

\[
\psi(X) = \inf_{A \in S_N^+} \phi(X, A), \quad \text{and} \quad A^*(X) := \arg\min_{A \in S_N^+} \phi(X, A) = (XX^\top + \lambda I_N)^{-1/2}.
\]

Substituting the variational definition of \( \psi \) into (30), we immediately get that \( X_t = Z_t \). Using the formula for \( A^*(X) \), we also get \( A_t = (X_tX_t^\top + \lambda I_N)^{-1/2} = M_{t-1} \). In addition, we have

\[
Z_t = \arg\min_{Z \in \mathcal{B}_{S(1)}} \phi(Z, A_t) + \langle \eta_{t-1} H_{t-1}, Z \rangle
\]

\[
= \arg\min_{Z \in \mathcal{B}_{S(1)}} \frac{1}{2} \left( \text{Tr}(A_t(ZZ^\top + \lambda I_N)) + \text{Tr}(A_t^{-1}) \right) + \langle \eta_{t-1} H_{t-1}, Z \rangle
\]

\[
= \arg\min_{Z \in \mathcal{B}_{S(1)}} \frac{1}{2} \text{Tr}(M_{t-1} ZZ^\top) + \langle \eta_{t-1} H_{t-1}, Z \rangle
\]

\[= Y_t. \]

These different writings then allow for different analyses of Schatten-1 FTRL.

**Analysis of** (30). In order to analyze update (30), we first need to symmetrize the iterate — the strong convexity results from Juditsky and Nemirovski (2008) only apply to symmetric matrices. Let \( X_{\text{sym}} = \begin{pmatrix} 0 & X \\ X^\top & 0 \end{pmatrix} \in S_{N+d} \), and \( H_{t-1, \text{sym}} = \begin{pmatrix} 0 & H_{t-1} \\ H_{t-1}^\top & 0 \end{pmatrix} \in S_{N+d} \). It is immediate to
We therefore focus on the regularizer \( F \). We can apply \( \text{Juditsky and Nemirovski, 2008, Proposition 3.1} \), with \( \Delta = [-1, 1] \) and \( f(x) = \sqrt{x^2 + \lambda} \), such that \( f''(x) = \lambda/(x^2 + \lambda)^{3/2} \) is lower bounded by \( \lambda/(1 + \lambda)^{3/2} \) over \( \Delta \). We get that \( F \) is \( \lambda/(1 + \lambda)^{3/2} \)-strongly convex with respect to the Frobenius norm. Applying \( \text{Orabona, 2019, Corollary 7.9} \), and returning to the non-symmetric formulation, we have for all \( U \in B_{\lambda}^{(1)} \)

\[
R_T(U) \leq \frac{\text{Tr}((UU^T + \lambda I_N)^{1/2}) - \lambda \sqrt{N}}{\eta_T} + \frac{(1 + \lambda)^{3/2}}{2\lambda} \sum_{t=1}^{T} \eta_{t-1} \|G_t\|_{F_t}^2
\]

\[
\leq 2L\sqrt{T} \left( \frac{\text{Tr}((UU^T + \lambda I_N)^{1/2}) - \lambda \sqrt{N}}{\lambda} \right)
\]

\[
\leq 2L\sqrt{T} \left( \frac{(1 + \lambda)^{3/2}}{\lambda} \right)
\]

\[
\leq 4L\sqrt{T}
\]

after optimizing in \( \eta_t \) and \( \lambda \). This bound is however worse than \( B \) by a factor \( \sqrt{N} \) —it is essentially equivalent to that of Euclidean FTRL. The main reason for this is that the analysis uses the Frobenius norm, that is unable to exploit the stochasticity of the activations.

**Analysis of \( B \).** On the contrary, update \( B \) is expressed in terms of Mahalanobis norms, that are crucial to the derivation of the target regret bound \( A \). Indeed, if we apply the FTRL analysis to the sequence of regularizers \( \psi_t(Y) = (1/2)\text{Tr}(M_{t-1}YY^T) \), which are \( 1 \)-strongly convex with respect to \( \| \cdot \|_{M_{t-1}} \), whose dual norms are the \( \| \cdot \|_{M_{t-1}^{-1}} \), we obtain that for all \( U \in \tilde{B}_{\lambda}^{(1)} \), we have

\[
\mathbb{E}[R_T(U)] \leq \frac{\text{Tr}(M_{t-1}UU^T)}{2\eta_T} + \frac{1}{2} \sum_{t=1}^{T} \eta_{t-1} \mathbb{E} \left[ \|M_t\|_{M_{t-1}^{-1}}^2 \right]
\]

\[
\leq \frac{\|M_{t-1}\|_2 \|U\|_{S(4)}^2}{2\eta_T} + \frac{L^2}{2\lambda} \sum_{t=1}^{T} \eta_{t-1} \text{Tr}(M_{t-1}^{-1})
\]

\[
\leq \frac{1}{2\eta_T} \sqrt{\sum_{i=1}^{N} \frac{1}{\sigma_i^2(X_T) + \lambda}} + \frac{L^2(1 + \sqrt{\lambda}N)}{2\lambda} \sum_{t=1}^{T} \eta_{t-1}
\]

\[
\leq \frac{1}{2\eta_T} \sqrt{\frac{N}{\lambda}} + \frac{L^2(1 + \sqrt{\lambda}N)}{2\lambda} \sum_{t=1}^{T} \eta_{t-1}
\]

\[
\leq L\sqrt{2T} \sqrt{\frac{1 + \sqrt{\lambda}N}{\lambda N}}
\]

\[
\leq L\sqrt{2TN^{1/4}}
\]

after optimizing in \( \eta_t \) and \( \lambda \). This bound is still not satisfactory as we have \( N^{1/4} \) instead of \( N^{-1/2} \). The problem here comes from the Bregman potential. We have \( \text{Tr}(M_{t-1}UU^T) \) that is difficult to upper bound. We would instead have liked \( \text{Tr}((UU^T + \lambda I_N)^{1/2}) \), but we do not know how close \( M_{t-1} \) is from \( (UU^T)^{-1/2} \). With the augmented regularizer approach, we trade the possibility of choosing \( M_{t-1} \) (recall that we include the interaction matrix in the comparator) against an additional term in the regret accounting for the learning of \( U \). We can then make it of the same order as the terms accounting for the learning of \( U \) by a suitable tuning of the parameters.

**Analysis of \( (32) \).** It is carried out in the core text. It constitutes the best of both worlds: it benefits both from the nice potential of \( (30) \) and from the Mahalanobis norms of \( (31) \).
B.2 A dual approach to multitask online learning

Consider the potential

$$\phi'(X, A) = \text{Tr}(AX^\top X) + \text{Tr}(A) + \text{Tr}(A^{-1})$$

defined over $\mathbb{R}^{N \times d} \times S^+_d$, see e.g., Argyriou et al. (2008, Section 4). Potential $\phi'$ differs from $\phi$, recall (10), as it features $\text{Tr}(AX^\top X)$ instead of $\text{Tr}(AXX^\top)$ — the domain of definition for $A$ is also different. We now show that using potential $\phi'$ does not allow to derive Theorem 1.

For simplicity, we can only focus on the Mahalanobis potential $\psi_A(X) = \text{Tr}(AX^\top X)$. Indeed, the main role of the additive terms $\text{Tr}(A)$ and $\text{Tr}(A^{-1})$ is to devise an algorithm whose regret scales as $\inf_{A \in S^+_d \times B} R_{T,A}$, where $R_{T,A}$ is the regret of the algorithm run with $\psi_A$, and we show that even with the best $A$ one cannot achieve bound (9). Using the FTRL analysis, we get

$$\forall U \in B_{S(1)}, \quad R_{T,A}(U) \leq \frac{\text{Tr}(AU^\top U)}{\eta} + \frac{\eta}{4} \sum_{t=1}^T \|G_t\|_{A^{-1}}^2,$$

with the following important notation change

$$\|G_t\|_{A^{-1}}^2 = \text{Tr}(A^{-1}G_t^\top G_t) = \sum_{i,j=1}^d A^{-1}_{ij} \langle [G_t]_{ij}, [G_t]_{ij} \rangle.$$  \hfill (34)

However, the row-sparsity of $G_t$ does not allow to simplify (34) further, and we can only use $\|G_t\|_{A^{-1}}^2 \leq \lambda_{\max}(A^{-1}) \|G_t\|_{F^2} \leq L^2 \lambda_{\max}(A^{-1})$. Substituting into (33) and optimizing with respect to $A$, we obtain a regret of $L\|U\|_2 \sqrt{T}$, which is looser than (9).

In order to make potential $\phi'$ really relevant, one should assume that the gradient matrices $G_t$ are column-sparse. Equation (34) then simplifies into $A_{d_i d_i}^{-1} \|g_t\|_2^2$, where $d_i$ is the active dimension at time step $t$ and $g_t \in \mathbb{R}^N$ is the non-zero column of $G_t$. Assuming stochastic dimension activations, the latter is bounded in expectation by $L^2 \text{Tr}(A^{-1})/d$, and one could achieve a regret of $L\|U\|_{S(1)} \sqrt{T}/d$, instead of $L\|U\|_{S(2)} \sqrt{NT}/d$ for Euclidean. However, although column-sparse gradients help the computations, the assumption of having stochastic dimension activations seems less applicable in practice than the one we studied.

C Additional experiments

We present additional synthetic experiments, with different task structures. We consider 4 settings: 2 orthogonal clusters, 3 orthogonal clusters, 2 opposite clusters (reported in the main text), and 4 opposite clusters. Precisely, we generate the task vectors in $\mathbb{R}^{10}$ as follows. We choose 2 orthogonal (respectively, 3 orthogonal, 2 opposite, 4 pairwise opposite, pairwise orthogonal) directions, and generate unit task vectors along these directions, perturbed by a centered Gaussian noise with standard deviation 0.05. The matrices $U$ are then rescaled such that $\|U\|_{S(1)} = 1$. The heat maps in Figure 3 (left column) represent the task covariance matrices $UU^\top$. The inputs $x_t$ are generated uniformly on the sphere, and the labels are $y_t = \langle u_{x_t}, x_t \rangle + \epsilon_t$, where $\epsilon_t \sim U[-0.01, 0.01]$. The values of $N$ and $T$ have been chosen to be as small as possible, but large enough to showcase the asymptotic behaviours of the algorithms: $N = 200$ and $T = 100, 000$ (i.e., 500 observations per task) for the 2 orthogonal clusters, $N = 100$ and $T = 100, 000$ (i.e., 1000 observations per task) for the 3 orthogonal clusters, $N = 100$ and $T = 10, 000$ (i.e., 100 observations per task) for the 2 opposite clusters, $N = 100$ and $T = 20, 000$ (i.e., 200 observations per task) for the 4 orthogonal clusters. We ran the algorithms with the square loss. The plots in Figure 3 (right column) show the cumulative square losses against time for Schatten-1 FTRL, Independent FTRL, and Variance FTRL.
Figure 3: Task covariance matrices and cumulative square losses for different task structures.
D Additional algorithms and figures

Algorithm 2 Schatten-1 FTRL

\textbf{input} : Parameter \( \lambda > 0 \), learning rates \((\eta_t)_{t \geq 0}\)
\textbf{init} : \( H = 0_{\mathbb{R}^N \times d} \)
\textbf{for} \( t = 1 \) to \( T \) do
  \quad Compute \( X_t = \argmin_{X \in B_{\mathbb{R}^N}} \text{Tr}\left( (XX^T + \lambda I_N)^{1/2} \right) + \langle \eta_{t-1} H, X \rangle \)
  \quad Receive \( i_t \) drawn uniformly at random from \( \{1, \ldots, N\} \)
  \quad Play \( x_t = [X_t]_{i_t} \)
  \quad Pay \( \ell_t(x_t) \) and receive \( g_t \in \partial \ell_t(x_t) \)
  \quad Update \( H: [H]_{i_t} \leftarrow [H]_{i_t} + g_t \)

Algorithm 3 Krichevsky-Trofimov for 1-dimensional OCO (Orabona, 2019, e.g.)

\textbf{for} \( t = 1 \) to \( T \) do
  \quad Play \( x_t = -\sum_{i=1}^{t-1} g_i \left( 1 - \sum_{i=1}^{t-1} s_i x_i \right) \)
  \quad Pay \( \ell_t(x_t) \) and receive \( g_t \in \partial \ell_t(x_t) \)

Algorithm 4 AdaTask for arbitrary activation probabilities

\textbf{input} : Number of tasks \( N \), activations probabilities \( \pi_1, \ldots, \pi_N \), parameter \( \lambda > 0 \), learning rates \((\eta_t)_{t \geq 0}\), Lipschitz constant \( L \)
\textbf{init} : \( H = 0_{\mathbb{R}^N \times d} \)
\textbf{for} \( t = 1 \) to \( T \) do
  \quad Compute \( \tilde{X}_t = \argmin_{X \in B_{\mathbb{R}^N}} \text{Tr}\left( \left( \Pi^{1/2} X X^T \Pi^{1/2} + \lambda I_N \right)^{1/2} \right) + \langle \eta_{t-1} H, X \rangle \)
  \quad Compute \( y_t = -\frac{1}{t} \sum_{i=1}^{t-1} s_i \left( 1 - \sum_{i=1}^{t-1} s_i y_i \right) \)
  \quad Compute \( X_t = y_t \tilde{X}_t \)
  \quad Receive \( i_t \) drawn uniformly at random from \( \{1, \ldots, N\} \)
  \quad Predict \( x_t = [X_t]_{i_t} \)
  \quad Pay \( \ell_t(x_t) \) and receive \( g_t \in \partial \ell_t(x_t) \)
  \quad Update \( H: [H]_{i_t} \leftarrow [H]_{i_t} + g_t \)
  \quad Set \( s_t = \frac{1}{L} \langle g_t, [\tilde{X}_t]_{i_t} \rangle \)

Figure 4: Function \( \sigma \mapsto \sqrt{\sigma^2 + \lambda} \) for different values of \( \lambda \).