Proceedings of Machine Learning Research vol 75:1–37, 2018 31st Annual Conference on Learning Theory

Black-Box Reductions for Parameter-free Online Learning in Banach Spaces

Ashok Cutkosky ASHOKC@CS.STANFORD.EDU
Stanford University

Francesco Orabona FRANCESCO@ORABONA.COM
Stony Brook University

Editors: Sebastien Bubeck, Vianney Perchet and Philippe Rigollet

Abstract

We introduce several new black-box reductions that significantly improve the design of adaptive and parameter-free online learning algorithms by simplifying analysis, improving regret guarantees, and sometimes even improving runtime. We reduce parameter-free online learning to online exp-concave optimization, we reduce optimization in a Banach space to one-dimensional optimization, and we reduce optimization over a constrained domain to unconstrained optimization. All of our reductions run as fast as online gradient descent. We use our new techniques to improve upon the previously best regret bounds for parameter-free learning, and do so for arbitrary norms.

1. Parameter Free Online Learning

Online learning is a popular framework for understanding iterative optimization algorithms, including stochastic optimization algorithms or algorithms operating on large data streams. For each of T iterations, an online learning algorithm picks a point w_t in some space W, observes a loss function $\ell_t : W \to \mathbb{R}$, and suffers loss $\ell_t(w_t)$. Performance is measured by the *regret*, which is the total loss suffered by the algorithm in comparison to some benchmark point $\mathring{w} \in W$:

$$R_T(\mathring{w}) = \sum_{t=1}^{T} \ell_t(w_t) - \ell_t(\mathring{w}) \ .$$

We want to design algorithms that guarantee low regret, even in the face of adversarially chosen ℓ_t.

To make the problem more tractable, we suppose W is a convex set and each ℓ_t is convex (this is called Online Convex Optimization). With this assumption, we can further reduce the problem to online *linear* optimization (OLO) in which each ℓ_t must be a linear function. To see the reduction, suppose g_t is a subgradient of ℓ_t at w_t ($g_t \in \partial \ell_t(w_t)$). Then $\ell_t(w_t) - \ell_t(\mathring{w}) \leq \langle g_t, w_t - \mathring{w} \rangle$, which implies $R_T(\mathring{w}) \leq \sum_{t=1}^{T} \langle g_t, w_t - \mathring{w} \rangle$. Our algorithms take advantage of this property by accessing ℓ_t only through g_t and controlling the linearized regret $\sum_{t=1}^{T} \langle g_t, w_t - \mathring{w} \rangle$.

Lower bounds for unconstrained online linear optimization [18; 21] imply that when ℓ_t are L-Lipschitz, no algorithm can guarantee regret better than $\Omega(\|\mathring{w}\| L \sqrt{T \ln(\|\mathring{w}\| LT + 1)})$. Relaxing the L-Lipschitz restriction on the losses leads to catastrophically bad lower bounds [5], so in this paper we focus on the case where a Lipschitz bound is known, and assume $L = 1$ for simplicity.[1]

1. One can easily rescale the g_t by L to incorporate arbitrary L.

Our primary contribution is a series of three reductions that simplify the design of *parameter-free* algorithms,[2] that is algorithms whose regret bound is optimal without the need to tune parameters (e.g. learning rates). First, we show that algorithms for online exp-concave optimization imply parameter-free algorithms for OLO (Section 2). Second, we show a general reduction from online learning in arbitrary dimensions with any norm to one-dimensional online learning (Section 3). Finally, given any two convex sets $W \subset V$, we construct an online learning algorithm over W from an online learning algorithm over V (Section 4).

All of our reductions are very general. We make no assumptions about the inner workings of the base algorithms and are able to consider any norm, so that W may be a subset of a Banach space rather than a Hilbert space or $\mathbb{R}^d$. Each reduction is of independent interest, even for non-parameter-free algorithms, but by combining them we can produce powerful new algorithms.

First, we use our reductions to design a new parameter-free algorithm that improves upon the prior regret bounds, achieving

$$R_T(\mathring{w}) \leq \|\mathring{w}\| \sqrt{\sum_{t=1}^{T} \|g_t\|_\star^2 \ln\left(\|\mathring{w}\| \sum_{t=1}^{T} \|g_t\|_\star^2 + 1\right)},$$

where $\|\cdot\|$ is any norm and $\|\cdot\|_\star$ is the dual norm ($\|g_t\|_\star = \|g_t\|$ when $\|\cdot\|$ is the 2-norm). Previous parameter-free algorithms [18; 20; 22; 23; 8; 5; 24] obtain at best an exponent of 1 in their dependence on $\|g_t\|_\star$ (which is worse because $\|g_t\|_\star \leq 1$ by our 1-Lipschitz assumption). Achieving $\|g_t\|_\star^2$ rather than $\|g_t\|_\star$ can imply asymptotically lower regret when the losses ℓ_t are smooth [27], so this is not merely a cosmetic difference. In addition to the worse regret bound, all prior analyses we are aware of are quite complicated, often involving pages of intricate algebra, and are usually limited to the 2-norm. In contrast, our techniques are both simpler and more general.

We further demonstrate the power of our reductions through three more applications. In Section 5, we consider the multi-scale experts problem studied in [9; 1] and improve prior regret guarantees and runtimes. In Section 6, we create an algorithm obtaining $\tilde{O}(\sqrt{T})$ regret for general convex losses, but logarithmic regret for strongly-convex losses using only first-order information, similar to [30; 7], but with runtime improved to match gradient descent. Finally, in Section 7 we prove a regret bound of the form $R_T(\mathring{w}) = \tilde{O}\left(\sqrt{d \sum_{t=1}^{T} \langle g_t, \mathring{w}\rangle^2}\right)$ for d-dimensional Banach spaces, extending the results of [14] to unconstrained domains. We summarize our results in Figure 1.

Notation. The dual of a Banach space B over a field F, denoted $B^\star$, is the set of all continuous linear maps $B \to F$. We will use the notation $\langle v, w\rangle$ to indicate the application of a dual vector $v \in B^\star$ to a vector $w \in B$. $B^\star$ is also a Banach space with the *dual norm*: $\|v\|_\star = \sup_{w \in B,\, \|v\|=1} \langle w, v\rangle$. For completeness, in Appendix A we recall some more background on Banach spaces.

2. Online Newton Step to Online Linear Optimization via Betting Algorithms

In this section we show *how to use the Online Newton Step (ONS) algorithm [12] to construct a 1D parameter-free algorithm*. Our approach relies on the coin-betting abstraction [23] for the design of parameter-free algorithms. Coin betting strategies record the *wealth* of the algorithm, which is

2. The name "parameter-free" was first used by Chaudhuri et al. [4] for an expert algorithm that does not need to know the entropy of the competitor to achieve the optimal regret bound for any competitor.

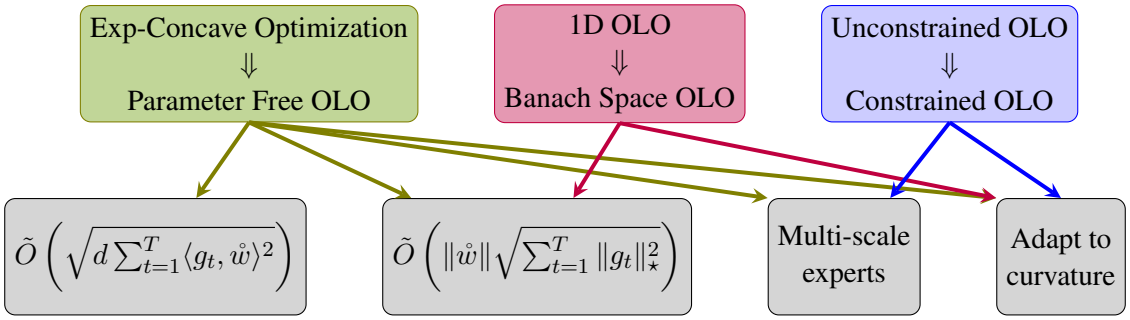

Figure 1: We prove three reductions (top row), and use these reductions to obtain specific algorithms and regret bounds (bottom row). Arrows indicate which reductions are used in each algorithm.

defined by some initial (i.e. user-specified) ϵ plus the total "reward" $\sum_{t=1}^{T} -g_t w_t$ it has gained:

$$\text{Wealth}_T = \epsilon - \sum_{t=1}^{T} g_t w_t \,. \tag{1}$$

Given this wealth measurement, coin betting algorithms "bet" a signed fraction $v_t \in (-1, 1)$ of their current wealth on the outcome of the "coin" $g_t \in [-1, 1]$ by playing $w_t = v_t \text{Wealth}_{T-1}$, so that $\text{Wealth}_T = \text{Wealth}_{T-1} - g_t v_t \text{Wealth}_{T-1}$. The advantage of betting algorithms lies in the fact that high wealth is equivalent to a low regret [20], but lower-bounding the wealth of an algorithm is conceptually simpler than upper-bounding its regret because the competitor $\mathring{w}$ does not appear in (1). Thus the question is how to pick betting fractions v_t that guarantee high wealth. This is usually accomplished through careful design of bespoke potential functions and meticulous algebraic manipulation, but we take a different and simpler path.

At a high level, our approach is to re-cast the problem of choosing betting fractions v_t as itself an online learning problem. We show that this online learning problem has *exp-concave* losses rather than linear losses. Exp-concave losses are known to be much easier to optimize than linear losses and it is possible to obtain $\ln(T)$ regret rather than the $\sqrt{T}$ limit for linear optimization [12]. So by using an exp-concave optimization algorithm such as the Online Newton Step (ONS), we find the optimal betting fraction $\mathring{v}$ very quickly, and obtain high wealth. The pseudocode for the resulting strategy is in Algorithm 1.

Later (in Section 7), we will see that this same 1D argument holds seamlessly in Banach spaces, where now the betting fraction v_t is a vector in the Banach space and the outcome of the coin g_t is a vector in the dual space with norm bounded by 1. We therefore postpone computing exact constants for the Big-O notation in Theorem 1 to the more general Theorem 8.

It is important to note that ONS in 1D is extremely simple to implement. Even the projection onto a bounded set becomes just a truncation between two real numbers, so that Algorithm 1 can run quickly. We can show the following regret guarantee:

Theorem 1 *For $|g_t| \leq 1$, Algorithm 1, guarantees the regret bound:*

$$R_T(\mathring{w}) = O\left[\epsilon + \max\left(|\mathring{w}| \ln\left[\frac{|\mathring{w}|\sum_{t=1}^{T} g_t^2}{\epsilon}\right] \,,\, |\mathring{w}|\sqrt{\sum_{t=1}^{T} g_t^2 \ln\left[\frac{|\mathring{w}|^2\sum_{t=1}^{T} g_t^2}{\epsilon^2} + 1\right]}\right)\right] .$$

Algorithm 1 Coin-Betting through ONS

Require: Initial wealth $\epsilon > 0$

Initialize: $\text{Wealth}_0 = \epsilon$, initial betting fraction $v_1 = 0$
for $t = 1$ **to** T **do**
  Bet $w_t = v_t \text{Wealth}_{t-1}$, Receive $g_t \in [-1, 1]$
  Update $\text{Wealth}_t = \text{Wealth}_{t-1} - g_t w_t$
  //compute new betting fraction $v_{t+1} \in [-1/2, 1/2]$ via ONS update on losses $-\ln(1 - g_t v)$
  Set $z_t = \frac{d}{dv_t}\left(-\ln(1 - g_t v_t)\right) = \frac{g_t}{1 - g_t v_t}$
  Set $A_t = 1 + \sum_{i=1}^{t} z_i^2$
  $v_{t+1} = \max\left(\min\left(v_t - \frac{2}{2-\ln(3)}\frac{z_t}{A_t}, 1/2\right), -1/2\right)$
end for

Proof Define $\text{Wealth}_T(\mathring{v})$ to be wealth of the betting algorithm that bets the constant (signed) fraction $\mathring{v}$ on every round, starting from initial wealth $\epsilon > 0$.

We begin with the regret-reward duality that is the start of all coin-betting analyses [23]. Suppose that we obtain a bound $\text{Wealth}_T \geq f_T\left(-\sum_{t=1}^{T} g_t\right)$ for some f_T. Then,

$$R_T(\mathring{w}) - \epsilon = -\text{Wealth}_T - \sum_{t=1}^{T} g_t \mathring{w} \leq -\sum_{t=1}^{T} g_t \mathring{w} - f_T\left(-\sum_{t=1}^{T} g_t\right) \leq \sup_{G \in \mathbb{R}} G\mathring{w} - f_T(G) = f_T^\star(\mathring{w}),$$

where $f_T^\star$ indicates the Fenchel conjugate, defined by $f_T^\star(x) = \sup_\theta \ \theta x - f_T(\theta)$.

So, now it suffices to prove a wealth lower bound. First, observing that $\text{Wealth}_T = \text{Wealth}_{T-1} - \text{Wealth}_{T-1} g_t v_t$, we derive a simple expression for $\ln \text{Wealth}_T$ by recursion:

$$\ln \text{Wealth}_T = \ln\left(\text{Wealth}_{T-1}(1 - g_t v_t)\right) = \ln(\epsilon) + \sum_{t=1}^{T} \ln(1 - v_t g_t)\ .$$

Similarly, we have $\ln \text{Wealth}_T(\mathring{v}) = \ln(\epsilon) + \sum_{t=1}^{T} \ln(1 - \mathring{v} g_t)$. We subtract the identities to obtain

$$\ln \text{Wealth}_T(\mathring{v}) - \ln \text{Wealth}_T = \sum_{t=1}^{T} -\ln(1 - v_t g_t) - \left(-\ln(1 - \mathring{v} g_t)\right)\ . \tag{2}$$

Now, the key insight of this analysis: we interpret equation (2) as the regret of an algorithm playing v_t on losses $\ell_t(v) = -\ln(1 - v g_t)$, so that we can write

$$\ln \text{Wealth}_T = \ln \text{Wealth}_T(\mathring{v}) - R_T^v(\mathring{v}), \tag{3}$$

where $R_T^v(\mathring{v})$ is the regret of our method for choosing v_t.

For the next step, observe that $-\ln(1 - g_t v)$ is exp-concave (a function f is exp-concave if $\exp(-f)$ is concave), so that choosing v_t is an online exp-concave optimization problem. Prior work on exp-concave optimization allows us to obtain $R_T^v(\mathring{v}) = O\left(\ln\left(\sum_{t=1}^{T} g_t^2\right)\right)$ for any $|\mathring{v}| \leq \frac{1}{2}$ using the ONS algorithm. Therefore (dropping all constants for simplicity), we use (3) to obtain $\text{Wealth}_T \geq \text{Wealth}_T(\mathring{v}) / \sum_{t=1}^{T} g_t^2$ for all $|\mathring{v}| \leq \frac{1}{2}$.

Finally, we need to show that there exists $\mathring{v}$ such that $\text{Wealth}_T(\mathring{v})/\sum_{t=1}^T g_t^2$ is high enough to guarantee low regret on our original problem. Consider $\mathring{v} = \frac{-\sum_{t=1}^T g_t}{2\sum_{t=1}^T g_t^2 + 2\left|\sum_{t=1}^T g_t\right|} \in [-1/2, 1/2]$. Then, we invoke the tangent bound $\ln(1+x) \geq x - x^2$ for $x \in [-1/2, 1/2]$ (e.g. see [2]) to see:

$$\ln \text{Wealth}_T(\mathring{v}) - \ln(\epsilon) = \sum_{t=1}^T \ln(1 - g_t\mathring{v}) \geq -\sum_{t=1}^T g_t\mathring{v} - \sum_{t=1}^T (g_t\mathring{v})^2 \geq \frac{\left(\sum_{t=1}^T g_t\right)^2}{4\sum_{t=1}^T g_t^2 + 4\left|\sum_{t=1}^T g_t\right|} .$$

$$\text{Wealth}_T \geq \epsilon \exp\left[\frac{\left(\sum_{t=1}^T g_t\right)^2}{4\sum_{t=1}^T g_t^2 + 4\left|\sum_{t=1}^T g_t\right|}\right] \Big/ \sum_{t=1}^T g_t^2 = f_T\left(\sum_{t=1}^T g_t\right) ,$$

where $f_T(x) = \epsilon \exp[x^2/(4\sum_{t=1}^T g_t^2 + 4|x|)]/\sum_{t=1}^T g_t^2$. To obtain the desired result, we recall that $\text{Wealth}_T \geq f_T\left(\sum_{t=1}^T g_t\right)$ implies $R_T(\mathring{w}) \leq \epsilon + f_T^\star(\mathring{w})$, and calculate $f_T^\star$ (see Lemma 19).

In order to implement the algorithm, observe that our reference betting fraction $\mathring{v}$ lies in $[-1/2, 1/2]$, so we can run ONS restricted to the domain $[-1/2, 1/2]$. Exact constants can be computed by substituting the constants coming from the ONS regret guarantee, as we do in Theorem 8. ■

3. From 1D Algorithms to Dimension-Free Algorithms

A common strategy for designing parameter-free algorithms is to first create an algorithm for 1D problems (as we did in the previous section), and then invoke some particular algorithm-specific analysis to extend the algorithm to high dimensional spaces [23; 6; 20]. This strategy is unappealing for a couple of reasons. First, these arguments are often somewhat tailored to the algorithm at hand, and so a new argument must be made for a new 1D algorithm (indeed, it is not clear that any prior dimensionality extension arguments apply to our Algorithm 1). Secondly, all such arguments we know of apply only to Hilbert spaces and so do not allow us to design algorithms that consider norms other than the standard Euclidean 2-norm. In this section we address both concerns by providing a *black-box reduction from optimization in any Banach space to 1D optimization*. In further contrast to previous work, our reduction can be proven in just a few lines.

Our reduction takes two inputs: an algorithm $\mathcal{A}_{\text{1D}}$ that operates with domain $\mathbb{R}$ and achieves regret $R_T^1(\mathring{w})$ for any $\mathring{w} \in \mathbb{R}$, and an algorithm $\mathcal{A}_S$ that operates with domain equal to the unit ball S in some Banach space B, $S = \{x \in B \ : \ \|x\| \leq 1\}$ and obtains regret $R_T^{\mathcal{A}_S}(\mathring{w})$ for any $\mathring{w} \in S$. In the case when B is $\mathbb{R}^d$ or a Hilbert space, then online gradient descent with adaptive step sizes can obtain $R_T^{\mathcal{A}_S}(\mathring{w}) = \sqrt{2\sum_{t=1}^T \|g_t\|_2^2}$ (which is independent of $\mathring{w}$) [13].

Given these inputs, the reduction uses the 1D algorithm $\mathcal{A}_{\text{1D}}$ to learn a "magnitude" z and the unit-ball algorithm $\mathcal{A}_S$ to learn a "direction" y. This direction and magnitude are multiplied together to form the final output $w = zy$. Given a gradient g, the "magnitude error" is given by $\langle g, y\rangle$, which is intuitively the component of the gradient parallel to w. The "direction error" is just g. Our reduction is described formally in Algorithm 2.

Theorem 2 *Suppose $\mathcal{A}_S$ obtains regret $R_T^{\mathcal{A}_S}(\mathring{w})$ for any competitor $\mathring{w}$ in the unit ball and $\mathcal{A}_{1D}$ obtains regret $R_T^1(\mathring{w})$ for any competitor $\mathring{w} \in \mathbb{R}$. Then Algorithm 2 guarantees regret:*

$$R_T(\mathring{w}) \leq R_T^1(\|\mathring{w}\|) + \|\mathring{w}\| R_T^{\mathcal{A}_S}(\mathring{w}/\|\mathring{w}\|) .$$

Where by slight abuse of notation we set $\mathring{w}/\|\mathring{w}\| = 0$ when $\mathring{w} = 0$. Further, the subgradients s_t sent to $\mathcal{A}_{1D}$ satisfy $|s_t| \leq \|g_t\|_\star$.

Algorithm 2 One Dimensional Reduction

Require: 1D Online learning algorithm $\mathcal{A}_{\text{1D}}$, Banach space B and Online learning algorithm $\mathcal{A}_S$ with domain equal to unit ball $S \subset B$

1: **for** $t = 1$ **to** T **do**
2: Get point $z_t \in \mathbb{R}$ from $\mathcal{A}_{\text{1D}}$
3: Get point $y_t \in S$ from $\mathcal{A}_S$
4: Play $w_t = z_t y_t \in B$, receive subgradient g_t
5: Set $s_t = \langle g_t, y_t \rangle$
6: Send s_t as the tth subgradient to $\mathcal{A}_{\text{1D}}$
7: Send g_t as the tth subgradient to $\mathcal{A}_S$
8: **end for**

Proof First, observe that $|s_t| \leq \|g_t\|_\star \|y_t\| \leq \|g_t\|_\star$ since $\|y_t\| \leq 1$ for all t. Now, compute:

$$\begin{aligned}
R_T(\mathring{w}) &= \sum_{t=1}^T \langle g_t, w_t - \mathring{w} \rangle = \sum_{t=1}^T \langle g_t, z_t y_t \rangle - \langle g_t, \mathring{w} \rangle \\
&= \sum_{t=1}^T \underbrace{\langle g_t, y_t \rangle z_t - \langle g_t, y_t \rangle \|\mathring{w}\|}_{\text{regret of } \mathcal{A}_{\text{1D}} \text{ at } \|\mathring{w}\| \in \mathbb{R}} + \langle g_t, y_t \rangle \|\mathring{w}\| - \langle g_t, \mathring{w} \rangle \\
&\leq R_T^1(\|\mathring{w}\|) + \|\mathring{w}\| \sum_{t=1}^T \underbrace{\langle g_t, y_t \rangle - \langle g_t, \mathring{w}/\|\mathring{w}\| \rangle}_{\text{regret of } \mathcal{A}_S \text{ at } \mathring{w}/\|w\| \in S} \\
&\leq R_T^1(\|\mathring{w}\|) + \|\mathring{w}\| R_T^{\mathcal{A}_S}(\mathring{w}/\|\mathring{w}\|),
\end{aligned}$$

■

With this reduction in hand, designing dimension-free and parameter-free algorithms is now exactly as easy as designing 1D algorithms, so long as we have access to a unit-ball algorithm $\mathcal{A}_S$. As mentioned, for any Hilbert space we indeed have such an algorithm. In general, algorithms $\mathcal{A}_S$ exist for most other Banach spaces of interest [28], and in particular one can achieve $R_T^{\mathcal{A}_S}(\mathring{w}) \leq O\left(\sqrt{\frac{1}{\lambda}\sum_{t=1}^T \|g_t\|_\star^2}\right)$ whenever B is $(2, \lambda)$-uniformly convex [25] using the Follow-the-Regularized-Leader algorithm with regularizers scaled by $\frac{\sqrt{\lambda}}{\sqrt{\sum_{i=1}^t \|g_i\|_\star^2}}$ [19].

Applying Algorithm 2 to our 1D Algorithm 1, for any $(2, \lambda)$-uniformly convex B, we obtain:

$$\begin{aligned}
R_T(\mathring{w}) = O\Bigg[\|\mathring{w}\| \max\left(\ln \frac{\|\mathring{w}\| \sum_{t=1}^T \|g_t\|_\star^2}{\epsilon}, \sqrt{\sum_{t=1}^T \|g_t\|_\star^2 \ln\left(\frac{\|\mathring{w}\|^2 \sum_{t=1}^T \|g_t\|_\star^2}{\epsilon^2} + 1 \right)} \right) \\
+ \frac{\|\mathring{w}\|}{\sqrt{\lambda}} \sqrt{\sum_{t=1}^T \|g_t\|_\star^2 + \epsilon} \Bigg] .
\end{aligned}$$

Spaces that satisfy this property include Hilbert spaces such as $\mathbb{R}^d$ with the 2-norm (in which case $\lambda = 1$), as well the $\mathbb{R}^d$ with the p-norm for $p \in (1, 2]$ (in which case $\lambda = p - 1$). Finally, observe

that the runtime of this reduction is equal to the runtime of $\mathcal{A}_{1\mathrm{D}}$ plus the runtime of $\mathcal{A}_S$, which in many cases (including $\mathbb{R}^d$ with 2-norm or Hilbert spaces) is the same as online gradient descent.

Not only does this provide the fastest known parameter-free algorithm for an arbitrary norm, it is also the first parameter-free algorithm to obtain a dependence on the gradients of $\|g_t\|_\star^2$ rather than $\|g_t\|_\star$[3]. This improved bound immediately implies much lower regret in easier settings, such as smooth losses with small loss values at $\mathring{w}$ [27].

4. Reduction to Constrained Domains

The previous algorithms have dealt with optimization over an entire vector space. Although common and important case in practice, sometimes we must perform optimization with constraints, in which each w_t and the comparison point $\mathring{w}$ must lie in some convex domain W that is not an entire vector space. This constrained problem is often solved with the classical Mirror Descent [31] or Follow-the-Regularized-Leader [26] analysis. However, these approaches have drawbacks: for unbounded sets, they typically maintain regret bounds that have suboptimal dependence on $\mathring{w}$, or, for bounded sets, they depend explicitly on the diameter of W. We will address these issues with a simple reduction. Given any convex domain $V \supset W$ and an algorithm $\mathcal{A}$ that maintains regret $R_T^{\mathcal{A}}(\mathring{w})$ for any $\mathring{w} \in V$, we obtain an algorithm that maintains $2R_T^{\mathcal{A}}(\mathring{w})$ for any $\mathring{w}$ in W.

Before giving the reduction, we define the distance to a convex set W as $S_W(x) = \inf_{d \in W} \|x - d\|$ as well as the projection to W as $\Pi_W(x) = \{d \in W : \|d - x\| \leq \|c - x\|, \forall c \in W\}$. Note that if B is reflexive,[4] $\Pi_W(x) \neq \emptyset$ and that it is a singleton if B is a Hilbert space [16, Exercise 4.1.4].

The intuition for our reduction is as follows: given a vector $z_t \in V$ from $\mathcal{A}$, we predict with any $w_t \in \Pi_W(z_t)$. Then give $\mathcal{A}$ a subgradient at z_t of the surrogate loss function $\langle g_t, \cdot \rangle + \|g_t\|_\star S_W$, which is just the original linearized loss plus a multiple of S_W. The additional term S_W serves as a kind of Lipschitz barrier that penalizes $\mathcal{A}$ for predicting with any $z_t \notin W$. Pseudocode for the reduction is given in Algorithm 3.

Algorithm 3 Constraint Set Reduction

Require: Reflexive Banach space B, Online learning algorithm $\mathcal{A}$ with domain $V \supset W \subset B$

for $t = 1$ **to** T **do**
  Get point $z_t \in V$ from $\mathcal{A}$
  Play $w_t \in \Pi_W(z_t)$, receive $g_t \in \partial \ell_t(w_t)$
  Set $\tilde{\ell}_t(x) = \frac{1}{2}\left(\langle g_t, x \rangle + \|g_t\|_\star S_W(x)\right)$
  Send $\tilde{g}_t \in \partial \tilde{\ell}_t(z_t)$ as tth subgradient to $\mathcal{A}$
end for

Theorem 3 *Assume that the algorithm $\mathcal{A}$ obtains regret $R_T^{\mathcal{A}}(\mathring{w})$ for any $\mathring{w} \in V$. Then Algorithm 3 guarantees regret:*

$$R_T(\mathring{w}) = \sum_{t=1}^{T} \langle g_t, w_t - \mathring{w} \rangle \leq 2R_T^{\mathcal{A}}(\mathring{w}), \quad \forall \mathring{w} \in W \,.$$

Further, the subgradients $\tilde{g}_t$ sent to $\mathcal{A}$ satisfy $\|\tilde{g}_t\|_\star \leq \|g_t\|_\star$.

3. Independently, [10] achieved the same runtime in the supervised prediction setting, but with no adaptivity to g_t.
4. All Hilbert spaces and finite-dimensional Banach spaces are reflexive.

Before proving this Theorem, we need a small technical Proposition, proved in Appendix D.

Proposition 1 *S_W is convex and 1-Lipschitz for any closed convex set W in a reflexive Banach space B.*

Proof [of Theorem 3] From Proposition 1, we observe that since S_W is convex and $\|g_t\|_\star \geq 0$, $\tilde{\ell}_t$ is convex for all t. Therefore, by $\mathcal{A}$'s regret guarantee, we have

$$\sum_{t=1}^{T} \tilde{\ell}_t(z_t) - \tilde{\ell}_t(\mathring{w}) \leq R_T^{\mathcal{A}}(\mathring{w}) \ .$$

Next, since $\mathring{w} \in W$, $\langle g_t, \mathring{w} \rangle = 2\tilde{\ell}_t(\mathring{w})$ for all t. Further, since $w_t \in \Pi_W(z_t)$, we have $\langle g_t, z_t \rangle + \|g_t\|_\star \|w_t - z_t\| = 2\tilde{\ell}_t(z_t)$. Finally, by the definition of dual norm we have

$$\langle g_t, w_t - \mathring{w} \rangle \leq \langle g_t, z_t - \mathring{w} \rangle + \|g_t\|_\star \|w_t - z_t\| = 2\tilde{\ell}_t(z_t) - 2\tilde{\ell}_t(\mathring{w}) \ .$$

Combining these two lines proves the regret bound of the theorem. The bound on $\|\tilde{g}_t\|_\star$ follows because S_W is 1-Lipschitz, from Proposition 1. ■

We conclude this section by observing that in many cases it is very easy to compute an element of Π_W and a subgradient of S_W. For example, when W is a unit ball, it is easy to see that $\Pi_W(x) = \frac{x}{\|x\|}$ and $\partial S_W(x) = \partial \|x\|$ for any x not in the ball. In general, we provide the following result that often simplifies computing the subgradient of S_W (proved in Appendix D):

Theorem 4 *Let B be a reflexive Banach space such that for every $0 \neq b \in B$, there is a unique dual vector $b^\star$ such that $\|b^\star\|_\star = 1$ and $\langle b^\star, b \rangle = \|b\|$. Let $W \subset B$ a closed convex set. Given $x \in B$ and $x \notin W$, let $p \in \Pi_W(x)$. Then $\{(x-p)^\star\} = \partial S_W(x)$.*

5. Reduction for Multi-Scale Experts

In this section, we apply our reductions to the multi-scale experts problem considered in [9; 1]. Our algorithm improves upon both prior algorithms: the approach of [1] has a mildly sub-optimal dependence on the prior distribution, while the approach of [9] takes time $O(T)$ per update, resulting in a quadratic total runtime. Our algorithm matches the regret bound of [9] while running in the same time complexity as online gradient descent.

The multi-scale experts problem is an online linear optimization problem over the probability simplex $\{x \in \mathbb{R}^N_{\geq 0} : \sum_{i=1}^N x_i = 1\}$ with linear losses $\ell_t(w) = g_t \cdot w$ such that each $g_t = (g_{t,1}, \ldots, g_{t,N})$ satisfies $|g_{t,i}| \leq c_i$ for some known quantities c_i. The objective is to guarantee that the regret with respect to the ith basis vector e_i (the ith "expert") scales with c_i. Formally, we want $R_T(\mathring{w}) = O(\sum_{i=1}^N c_i |\mathring{w}_i| \sqrt{T \log(c_i |\mathring{w}_i| T/\pi_i)})$, given a prior discrete distribution $(\pi_1, \ldots, \pi_N)$. As discussed in depth by [9], such a guarantee allows us to combine many optimization algorithms into one meta-algorithm that converges at the rate of the best algorithm *in hindsight*.

We accomplish this through two reductions. First, given any distribution $(\pi_1, \ldots, \pi_N)$ and any family of 1-dimensional OLO algorithms $\mathcal{A}(\epsilon)$ that guarantees $R(u) \leq O\left(\epsilon + |u| \sqrt{\log(|u|T/\epsilon)T}\right)$ on 1-Lipschitz losses for any given ϵ (such as our Algorithm 1 or many other parameter-free algorithms), we apply the classic "coordinate-wise updates" trick [29] to generate an N-dimensional OLO algorithm with regret $R_T(u) = O\left(\epsilon + \sum_{i=1}^N |u_i| \sqrt{\log\left(|u_i|T/(\epsilon\pi_i)\right) T}\right)$ on losses that are 1-Lipschitz with respect to the 1-norm.

Algorithm 4 Coordinate-Wise Updates

Require: parametrized family of 1-D online learning algorithm $\mathcal{A}(\epsilon)$, prior π, $\epsilon > 0$

Initialize: N copies of $\mathcal{A}$: $\mathcal{A}_1(\epsilon\pi_1), \dots, \mathcal{A}_N(\epsilon\pi_N)$
for $t = 1$ **to** T **do**
  Get points $z_{t,i}$ from $\mathcal{A}_i$ for all i to form vector $z_t = (z_{t,1}, \dots, z_{t,N})$
  Play z_t, get loss $g_t \in \mathbb{R}^N$ with $\|g_t\|_\infty \leq 1$
  Send $g_{t,i}$ to $\mathcal{A}_i$ for all i
end for

Theorem 5 *Suppose for any $\epsilon > 0$, $\mathcal{A}(\epsilon)$ guarantees regret*

$$R_T(u) \leq O\left(\epsilon + |u|\sqrt{\log\left(\frac{|u|T}{\epsilon} + 1\right)T}\right)$$

for 1-dimensional losses bounded by 1*. Then Algorithm 4 guarantees regret*

$$R_T(u) \leq O\left(\epsilon + \sum_{i=1}^N |u_i|\sqrt{\log\left(\frac{|u_i|T}{\epsilon\pi_i} + 1\right)T}\right) .$$

Proof Let $R_T^i(u_i)$ be the regret of the ith copy of $\mathcal{A}$ with respect to $u_i \in \mathbb{R}$. Then

$$\sum_{t=1}^T \langle g_t, w_t - u\rangle = \sum_{i=1}^N \sum_{t=1}^T g_{t,i}(w_{t,i} - u_i) \leq \sum_{i=1}^N R_T^i(u_i) \leq O\left(\epsilon + \sum_{i=1}^N |u_i|\sqrt{\log\left(\frac{|u_i|T}{\epsilon\pi_i} + 1\right)T}\right) .$$

■

Algorithm 5 Multi-Scale Experts

Require: parametrized 1-D Online learning algorithm $\mathcal{A}(\epsilon)$, prior π, scales $c_1, \dots, c_N$

Initialize: coordinate-wise algorithm $\mathcal{A}_\pi$ with prior π using $\mathcal{A}(\epsilon)$
Define $W = \{x : x_i \geq 0 \text{ for all } i \text{ and } \sum_{i=1}^N x_i/c_i = 1\}$
Let $\mathcal{A}_\pi^W$ be the result of applying the unconstrained-to-constrained reduction to $\mathcal{A}_\pi$ with constraint set W using $\|\cdot\|_1$
for $t = 1$ **to** T **do**
  Get point $z_t \in W$ from $\mathcal{A}_\pi^W$
  Set $x_t \in \mathbb{R}^N$ by $x_{t,i} = z_{t,i}/c_i$. Observe that x_t is in the probability simplex
  Play x_t, get loss vector g_t
  Set $\tilde{g}_t \in \mathbb{R}^N$ by $\tilde{g}_{t,i} = \frac{g_{t,i}}{c_i}$
  Send $\tilde{g}_t$ to $\mathcal{A}_\pi^W$
end for

With this in hand, notice that applying our reduction Algorithm 3 with the 1-norm easily yields an algorithm over the probability simplex W with the same regret (up to a factor of 2), as long as $\|g_t\|_\infty \leq 1$. Then, we apply an affine change of coordinates to make our multi-scale experts losses have $\|g_t\|_\infty \leq 1$, so that applying this algorithm yields the desired result (see Algorithm 5).

Theorem 6 *If g_t satisfies $|g_{t,i}| \leq c_i$ for all t and i and $\mathcal{A}(\epsilon)$ satisfies the conditions of Theorem 5, then, for any $\mathring{w}$ in the probability simplex, Algorithm 5 satisfies the regret bound*

$$R_T(\mathring{w}) \leq O\left(\epsilon + \sum_{i=1}^{N} c_i|\mathring{w}_i|\sqrt{\log\left(\frac{c_i|\mathring{w}_i|T}{\epsilon\pi_i} + 1\right)T}\right) .$$

Proof Given any $\mathring{w}$ in the probability simplex, define $\tilde{w} \in \mathbb{R}^N$ by $\tilde{w}_i = c_i\mathring{w}_i$. Observe that $\tilde{w} \in W$. Further, observe that since $|g_{t,i}| \leq c_i$, $\|\tilde{g}_t\|_\infty \leq 1$. Finally, observe that $\tilde{g}_t \cdot z_t = \sum_{i=1}^N \tilde{g}_{t,i}z_{t,i} = \sum_{i=1}^N \frac{g_{t,i}}{c_i}c_i x_{t,i} = g_t \cdot x_t$ and similarly $\tilde{g}_t \cdot \tilde{w} = g_t \cdot \mathring{w}$. Thus $\sum_{t=1}^T \tilde{g}_t \cdot z_t - \tilde{g}_t \cdot \tilde{w} = \sum_{t=1}^T g_t \cdot (x_t - \mathring{w})$. Now, by Theorem 5 and Theorem 3 we have

$$\sum_{t=1}^{T} g_t \cdot (x_t - \mathring{w}) = \sum_{t=1}^{T} \tilde{g}_t \cdot (z_t - \tilde{w}) \leq O\left(\epsilon + \sum_{i=1}^{N} |\tilde{w}_i|\sqrt{\log\left(\frac{|\tilde{w}_i|T}{\epsilon\pi_i} + 1\right)T}\right)$$

Now simply substitute the definition $\tilde{w}_i = c_i\mathring{w}_i$ to complete the proof. ■

In Appendix E we show how to compute the projection Π_S and a subgradient of S_W in $O(N)$ time via a simple greedy algorithm. As a result, our entire reduction runs in $O(N)$ time per update.

6. Reduction to Adapt to Curvature

In this section, we present a black-box reduction to make a generic online learning algorithm over a Banach space adaptive to the curvature of the losses. Given a set W of diameter $D = \sup_{x,y\in W} \|x - y\|$, our reduction obtains $O(\log(TD)^2/\mu)$ regret on online μ-strongly convex optimization problems, but still guarantees $O(\log(TD)^2 D\sqrt{T})$ regret for online linear optimization problems, both of which are only log factors away from the optimal guarantees. We follow the intuition of [7], who suggest adding a weighted average of previous w_ts to the outputs of a base algorithm as a kind of "momentum" term. We improve upon their regret guarantee by a log factor and by the $\|g_t\|_\star^2$ terms instead of $\|g_t\|_\star$. More importantly, their algorithm involves an optimization step which may be very slow for most domains (e.g. the unit ball). In contrast, thanks to our fast reduction in Section 4, we keep the same running time as the base algorithm. Finally, previous results for algorithms with similar regret (e.g. [7; 30]) show logarithmic regret only for *stochastic* strongly convex problems. We give a two-line argument extending this to the adversarial case as well.

Theorem 7 *Let $\mathcal{A}$ be an online linear optimization algorithm that outputs w_t in response to g_t. Suppose W is a convex closed set of diameter D. Suppose $\mathcal{A}$ guarantees for all t and $\mathring{v}$:*

$$\sum_{i=1}^{t}\langle \tilde{g}_i, w_i - \mathring{v}\rangle \leq \epsilon + \|\mathring{v}\|A\sqrt{\sum_{i=1}^{t}\|\tilde{g}_i\|_\star^2\left(1 + \ln\left(\frac{\|\mathring{v}\|^2 t^C}{\epsilon^2} + 1\right)\right)} + B\|\mathring{v}\|\ln\left(\frac{\|\mathring{v}\|t^C}{\epsilon} + 1\right),$$

for constants A, B and C and ϵ independent of t. Then for all $\mathring{w} \in W$, Algorithm 6 guarantees

$$R_T(\mathring{w}) \leq \sum_{t=1}^{T}\langle g_t, x_t - \mathring{w}\rangle \leq O\left(\sqrt{V_T(\mathring{w})\ln\frac{TD}{\epsilon}\ln(T)} + \ln\frac{DT}{\epsilon}\ln(T) + \epsilon\right),$$

where $V_T(\mathring{w}) := \|\overline{x}_0 - \mathring{w}\|^2 + \sum_{t=1}^T \|\tilde{g}_t\|_\star^2\|x_t - \mathring{w}\|^2 \leq D^2 + \sum_{t=1}^T \|g_t\|_\star^2\|x_t - \mathring{w}\|^2$.

Algorithm 6 Adapting to Curvature

Require: Online learning algorithm $\mathcal{A}$
Initialize: W, a convex closed set in a reflexive Banach space, $\overline{x}_0$ an arbitrary point in W
for $t = 1$ **to** T **do**
  Get point w_t from $\mathcal{A}$
  Set $z_t = w_t + \overline{x}_{t-1}$
  Play $x_t \in \Pi_W(z_t)$, receive subgradient $g_t \in \partial \ell_t(x_t)$
  Set $\tilde{g}_t \in g_t + \|g_t\|_\star \partial S_W(z_t)$
  Set $\overline{x}_t = \frac{\overline{x}_0 + \sum_{i=1}^t \|\tilde{g}_i\|_\star^2 x_i}{1 + \sum_{i=1}^t \|\tilde{g}_i\|_\star^2}$
  Send $\tilde{g}_t$ so $\mathcal{A}$ as the tth subgradient
end for

To see that Theorem 7 implies logarithmic regret on online strongly-convex problems, suppose that each ℓ_t is μ-strongly convex, so that $\ell_t(w_t) - \ell(\mathring{w}) \leq \langle g_t, w_t - \mathring{w} \rangle - \frac{\mu}{2}\|w_t - \mathring{w}\|^2$. Then:

$$\begin{aligned}\sum_{t=1}^T \ell(x_t) - \ell(\mathring{w}) &\leq O\left(\sqrt{\log^2(DT)\sum_{t=1}^T \|x_t - \mathring{w}\|^2} - \frac{\mu}{2}\sum_{t=1}^T \|x_t - \mathring{w}\|^2 + \log^2(TD)\right) \\ &\leq O\left(\sup_X \sqrt{\log^2(DT)X} - \frac{\mu}{2}X + \log^2(TD)\right) = O\left(\log^2(DT)\left(1 + \frac{1}{\mu}\right)\right) .\end{aligned}$$

Where we have used $\|g_t\|_\star \leq 1$.

7. Banach-space betting through ONS

In this section, we present the Banach space version of the one-dimensional Algorithm 1. The pseudocode is in Algorithm 7. We state the algorithm in its most general Banach space formulation, which obscures some of its simplicity in more common scenarios. For example, when B is $\mathbb{R}^d$ equipped with the p-norm, then the linear operator L can be taken to be simply the identity map $I : \mathbb{R}^d \to \mathbb{R}^d \cong (\mathbb{R}^d)^\star$, and the ONS portion of the algorithm is the standard d-dimensional ONS algorithm. We give the regret guarantee of Algorithm 7 in Theorem 8. The proof, modulo technical details of ONS in Banach spaces, is identical to Theorem 1, and can be found in Appendix C.

Theorem 8 *Let B be a d-dimensional real Banach space and $u \in B$ be an arbitrary unit vector. Then, there exists a linear operator L such that using the Algorithm 7, we have for any $\mathring{w} \in B$,*

$$R_T(\mathring{w}) \leq \epsilon + \max\left\{\frac{d\|\mathring{w}\|}{2} - 8\|\mathring{w}\| + 8\|\mathring{w}\| \ln\left[\frac{8\|\mathring{w}\|\left(1 + 4\sum_{t=1}^T \|g_t\|_\star^2\right)^{4.5d}}{\epsilon}\right], \right.$$
$$\left. 2\sqrt{\sum_{t=1}^T \langle g_t, \mathring{w}\rangle^2 \ln\left(\frac{5\|\mathring{w}\|^2}{\epsilon^2}\left(8\sum_{t=1}^T \|g_t\|^2 + 2\right)^{9d+1} + 1\right)}\right\} .$$

Algorithm 7 Banach-space betting through ONS

Require: Real Banach space B, initial linear operator $L : B \to B^\star$, initial wealth $\epsilon > 0$

Initialize: $\text{Wealth}_0 = \epsilon$, initial betting fraction $v_1 = 0 \in S = \{x \in B \ : \|x\| \leq \frac{1}{2}\}$
for $t = 1$ **to** T **do**
  Bet $w_t = v_t \text{Wealth}_{t-1}$, receive g_t, with $\|g_t\|_\star \leq 1$
  Update $\text{Wealth}_t = \text{Wealth}_{t-1} - \langle g_t, w_t \rangle$
  //compute new betting fraction $v_{t+1} \in S$ via ONS update on losses $-\ln(1 - \langle g_t, v \rangle)$:
  Set $z_t = \frac{d}{dv_t}\left(-\ln(1 - \langle g_t, v_t \rangle)\right) = \frac{g_t}{1 - \langle g_t, v_t \rangle}$
  Set $A_t(x) = L(x) + \sum_{i=1}^t z_i \langle z_i, x \rangle$
  $v_{t+1} = \Pi_S^{A_t}(v_t - \frac{2}{2-\ln(3)} A_t^{-1}(z_t))$, where $\Pi_S^{A_t}(x) = \text{argmin}_{y \in S} \langle A_t(y - x), y - x \rangle$
end for

The main particularity of this bound is the presence of the terms $\sqrt{d \sum_{t=1}^T \langle g_t, \mathring{w} \rangle^2}$ rather than the usual $\|\mathring{w}\| \sqrt{\sum_{t=1}^T \|g_t\|_\star^2}$. We can interpret this bound as being adaptive to any *sequence* of norms $\|\cdot\|_1, \dots, \|\cdot\|_t$ because $\sqrt{d \sum_{t=1}^T \langle g_t, \mathring{w} \rangle^2} \leq \sqrt{d \sum_{t=1}^T \|\mathring{w}\|_t^2 (\|g_t\|_t)_\star^2}$. A similar kind of "many norm adaptivity" was recently achieved in [9], which competes with the best *fixed* L_p norm (or the best fixed norm in any finite set). Our bound in Theorem 8 is a factor of $\sqrt{d}$ worse,[5] but we can compete with any possible sequence of norms rather than with any fixed one.

Similar regret bounds to our Theorem 8 have already appeared in the literature. The first one we are aware of is the Second Order Perceptron [3] whose *mistake bound* is exactly of the same form. Recently, a similar bound was also proven in [14], under the assumption that W is of the form $W = \{\mathring{v} : \langle g_t, \mathring{v} \rangle \leq C\}$, for a known C. Also, Kotłowski [15] proved the same bound when the losses are of the form $\ell_t(w_t) = \ell(y_t, w_t \cdot x_t)$ and the algorithm receives x_t before its prediction. In contrast, we can deal with unbounded W and arbitrary convex losses through the use of subgradients. Interestingly, all these algorithms (including ours) have a $O(d^2)$ complexity per update.

8. Conclusions

We have introduced a sequence of three reductions showing that parameter-free online learning algorithms can be obtained from online exp-concave optimization algorithms, that optimization in a vector space with any norm can be obtained from 1D optimization, and that online optimization with constraints is no harder than optimization without constraints. Our reductions result in simpler arguments in many cases, and also often provide better algorithms in terms of regret bounds or runtime. We therefore hope that these tools will be useful for designing new online learning algorithms.

Acknowledgments

This material is based upon work partly supported by the National Science Foundation under grant no. 1740762 "Collaborative Research: TRIPODS Institute for Optimization and Learning" and by a Google Research Award for FO.

5. The dependence on d is unfortunately unimprovable, as shown by [17].

References

[1] Sébastien Bubeck, Nikhil R Devanur, Zhiyi Huang, and Rad Niazadeh. Online auctions and multi-scale online learning. *arXiv preprint arXiv:1705.09700*, 2017.

[2] Nicolò Cesa-Bianchi and Gábor Lugosi. *Prediction, learning, and games*. Cambridge university press, 2006.

[3] Nicolò Cesa-Bianchi, Alex Conconi, and Claudio Gentile. A second-order Perceptron algorithm. *SIAM Journal on Computing*, 34(3):640–668, 2005.

[4] Kamalika Chaudhuri, Yoav Freund, and Daniel J Hsu. A parameter-free hedging algorithm. In *Advances in neural information processing systems*, pages 297–305, 2009.

[5] Ashok Cutkosky and Kwabena Boahen. Online learning without prior information. In Satyen Kale and Ohad Shamir, editors, *Proc. of the 2017 Conference on Learning Theory*, volume 65 of *Proc. of Machine Learning Research*, pages 643–677, Amsterdam, Netherlands, 07–10 Jul 2017. PMLR.

[6] Ashok Cutkosky and Kwabena A Boahen. Online convex optimization with unconstrained domains and losses. In *Advances in Neural Information Processing Systems 29*, pages 748–756, 2016.

[7] Ashok Cutkosky and Kwabena A Boahen. Stochastic and adversarial online learning without hyperparameters. In *Advances in Neural Information Processing Systems*, pages 5066–5074, 2017.

[8] Dylan J Foster, Alexander Rakhlin, and Karthik Sridharan. Adaptive online learning. In *Advances in Neural Information Processing Systems 28*, pages 3375–3383. 2015.

[9] Dylan J Foster, Satyen Kale, Mehryar Mohri, and Karthik Sridharan. Parameter-free online learning via model selection. In *Advances in Neural Information Processing Systems*, pages 6022–6032, 2017.

[10] Dylan J Foster, Alexander Rakhlin, and Karthik Sridharan. Online learning: Sufficient statistics and the burkholder method. *arXiv preprint arXiv:1803.07617*, 2018.

[11] Petr Hájek, Vicente Montesinos Santalucía, Jon Vanderwerff, and Václav Zizler. *Biorthogonal systems in Banach spaces*. Springer Science & Business Media, 2007.

[12] Elad Hazan, Amit Agarwal, and Satyen Kale. Logarithmic regret algorithms for online convex optimization. *Machine Learning*, 69(2-3):169–192, 2007.

[13] Elad Hazan, Alexander Rakhlin, and Peter L Bartlett. Adaptive online gradient descent. In *Advances in Neural Information Processing Systems*, pages 65–72, 2008.

[14] Tomer Koren and Roi Livni. Affine-invariant online optimization and the low-rank experts problem. In *Advances in Neural Information Processing Systems 30*, pages 4750–4758. Curran Associates, Inc., 2017.

[15] Wojciech Kotłowski. Scale-invariant unconstrained online learning. In *Proc. of ALT*, 2017.

[16] Roberto Lucchetti. *Convexity and well-posed problems*. Springer Science & Business Media, 2006.

[17] Haipeng Luo, Alekh Agarwal, Nicolo Cesa-Bianchi, and John Langford. Efficient second order online learning by sketching. In *Advances in Neural Information Processing Systems*, pages 902–910, 2016.

[18] Brendan McMahan and Matthew Streeter. No-regret algorithms for unconstrained online convex optimization. In *Advances in neural information processing systems*, pages 2402–2410, 2012.

[19] H Brendan McMahan. A survey of algorithms and analysis for adaptive online learning. *Journal of Machine Learning Research (JMLR)*, 18(90):1–50, 2017.

[20] H Brendan McMahan and Francesco Orabona. Unconstrained online linear learning in hilbert spaces: Minimax algorithms and normal approximations. In *Conference on Learning Theory (COLT)*, pages 1020–1039, 2014.

[21] Francesco Orabona. Dimension-free exponentiated gradient. In *Advances in Neural Information Processing Systems*, pages 1806–1814, 2013.

[22] Francesco Orabona. Simultaneous model selection and optimization through parameter-free stochastic learning. In *Advances in Neural Information Processing Systems*, pages 1116–1124, 2014.

[23] Francesco Orabona and Dávid Pál. Coin betting and parameter-free online learning. In *Advances in Neural Information Processing Systems 29*, pages 577–585, 2016.

[24] Francesco Orabona and Tatiana Tommasi. Training deep networks without learning rates through coin betting. In *Advances in Neural Information Processing Systems*, pages 2157–2167, 2017.

[25] Iosif Pinelis. Rosenthal-type inequalities for martingales in 2-smooth banach spaces. *Theory Probab. Appl.*, 59(4):699–706, 2015.

[26] Shai Shalev-Shwartz. *Online learning: Theory, algorithms, and applications*. PhD thesis, Hebrew University, 2007.

[27] Nathan Srebro, Karthik Sridharan, and Ambuj Tewari. Smoothness, low noise and fast rates. In *Advances in Neural Information Processing Systems 23*, pages 2199–2207, 2010.

[28] Nathan Srebro, Karthik Sridharan, and Ambuj Tewari. On the universality of online mirror descent. In *Advances in neural information processing systems*, pages 2645–2653, 2011.

[29] Matthew Streeter and H Brendan McMahan. Less regret via online conditioning. *arXiv preprint arXiv:1002.4862*, 2010.

[30] Tim van Erven and Wouter M Koolen. MetaGrad: Multiple learning rates in online learning. In *Advances in Neural Information Processing Systems 29*, pages 3666–3674. 2016.

[31] Martin Zinkevich. Online convex programming and generalized infinitesimal gradient ascent. In *Proc. of the 20th International Conference on Machine Learning (ICML-03)*, pages 928–936, 2003.

Appendix

This appendix is organized as follows:

1. In Section A we collect some background information about Banach spaces, their duals, and other properties.
2. In Section B we provide an analysis of the ONS algorithm in Banach spaces that is useful for proving Theorem 8.
3. In Section C we apply this analysis of ONS in Banach spaces to prove Theorem 8, and provide the missing Fenchel conjugate calculation required to prove Theorem 1, which are our reductions from parameter-free online learning to Exp-concave optimization.
4. In Section D we prove Proposition 1, used in our reduction from constrained optimization to unconstrained optimization in Section 4. In this section we also prove Theorem 4, which simplifies computing subgradients of S_W in many cases.
5. In Section E we show how to compute Π_W and a subgradient of S_W on $O(N)$ time for use in our multi-scale experts algorithm.
6. Finally, in Section F we prove Theorem 7, our regret bound for an algorithm that adapts to stochastic curvature.

Appendix A. Banach Spaces

Definition 2 *A* Banach space *is a vector space B over $\mathbb{R}$ or $\mathbb{C}$ equipped with a norm $\|\cdot\| : B \to \mathbb{R}$ such that B is complete with respect to the metric $d(x, y) = \|x - y\|$ induced by the norm.*

Banach spaces include the familiar vector spaces $\mathbb{R}^d$ equipped with the Euclidean 2-norm, as well as the the same vector spaces equipped with the p-norm instead.

An important special case of Banach spaces are the Hilbert spaces, which are Banach spaces that are also equipped with an inner-product $\langle , \rangle : B \times B \to \mathbb{R}$ (a symmetric, positive definite, non-degenerate bilinear form) such that $\langle b, b\rangle = \|b\|^2$ for all $b \in B$. In the complex case, the inner-product is $\mathbb{C}$ valued and the symmetric part of the definition is replaced with the condition $\langle v, w\rangle = \overline{\langle w, v\rangle}$ where $\overline{x}$ indicates complex conjugation. Hilbert spaces include the typical examples of $\mathbb{R}^d$ with the usual dot product, as well as reproducing kernel Hilbert spaces.

The dual of a Banach space B over a field F, denoted $B^\star$, is the set of all continuous linear functions $B \to F$. For Hilbert spaces, there is a natural isomorphism $B \cong B^\star$ given by $b \mapsto \langle b, \cdot\rangle$. Inspired by this isomorphism, in general we will use the notation $\langle v, w\rangle$ to indicate application of a dual vector $v \in B^\star$ to a vector $w \in B$. It is important to note that our use of this notation in no way implies the existence of an inner-product on B. When B is a Banach space, $B^\star$ is also a Banach space with the *dual norm*: $\|w\|_\star = \sup_{v \in B,\, \|v\|=1} \langle w, v\rangle$. A subgradient of a convex function $\ell : B \to \mathbb{R}$ is naturally an element of the dual $B^\star$. Therefore, the reduction to linear losses by $\ell_t(w_t) - \ell_t(\mathring{w}) \le \langle g_t, w_t - \mathring{w}\rangle$ for $g_t \in \partial \ell_t(w_t)$ generalizes perfectly to the case where W is a convex subset of a Banach space.

Given any vector space V, there is a natural injection $V \to V^{\star\star}$ given by $x \mapsto \langle \cdot, x\rangle$. When this injection is an isomorphism of Banach spaces, then the space V is called *reflexive*. All finite-dimensional Banach spaces are reflexive.

Given any linear map of Banach spaces $T : X \to Y$, we define the *adjoint* map $T^\star : Y^\star \to X^\star$ by $T^\star(y^\star)(x) = \langle y^\star, T(x)\rangle$. $T^\star$ has the property (by definition) that $\langle y^\star, T(x)\rangle = \langle T^\star(y^\star), x\rangle$. As a special case, if B is a reflexive Banach space and $T : B \to B^\star$, then we can use the natural identification between $B^{\star\star}$ and B to view $T^\star$ as $T^\star : B \to B^\star$. Thus, in this case it is possible to have $T = T^\star$, in which case we call T self-adjoint.

Definition 3 *We define a Banach space B as (p, D) uniformly convex if [25]:*

$$\|x + y\|^p + \|x - y\|^p \geq 2\|x\|^p + 2D\|y\|^p, \quad \forall x, y \in B\,. \tag{4}$$

From this definition, we can see that if B is $(2, D)$ uniformly convex, then $\|\cdot\|^2$ is a D-strongly convex function with respect to $\|\cdot\|$:

Lemma 9 *Let $f(x)$ a convex function that satisfies*

$$f\left(\frac{x+y}{2}\right) \leq \frac{1}{2}f(x) + \frac{1}{2}f(y) - \frac{D}{2p}\|x - y\|^p\,.$$

Then, f satisfies $f(x + \delta) \geq f(x) + g(\delta) + D\frac{\|\delta\|^p}{p}$ for any subgradient $g \in \partial f(x)$. In particular for $p = 2$, f is D strongly convex with respect to $\|\cdot\|$.

Proof Set $y = x + 2\delta$ for some arbitrary δ. Let $g \in \mathbb{X}^\star$ be an arbitrary subgradient of f at x. Let $R_x(\tau) = f(x + \tau) - (f(x) + g(\tau))$. Then

$$f(x) + g(\delta) \leq f\left(\frac{x+y}{2}\right) \leq \frac{f(x) + f(x + 2\delta)}{2} - \frac{D\|2\delta\|^p}{2p} = f(x) + g(\delta) + \frac{R_x(2\delta)}{2} - \frac{D\|2\delta\|^p}{2p},$$

that implies $\frac{D}{p}\|2\delta\|^p \leq R_x(2\delta)$. So that $f(x+\tau) = f(x)+g(\tau)+R_x(\tau) \geq f(x)+g(\tau)+\frac{D}{p}\|\tau\|^p$ as desired. ■

Lemma 10 *Let B be a $(2, D)$ uniformly convex Banach space, then $f(x) = \frac{1}{2}\|x\|^2$ is D-strongly convex.*

Proof Let $x = u + v$ and $y = u - v$. Then, from the definition of $(2, D)$ uniformly convex Banach space, we have

$$2\|u + v\|^2 + 2D\|u - v\|^2 \leq 4\|u\|^2 + 4\|v\|^2,$$

that is

$$\frac{1}{2}\left\|\frac{u+v}{2}\right\|^2 \leq \frac{1}{2}\|u\|^2 + \frac{1}{2}\|v\|^2 - \frac{D}{4}\|u - v\|^2\,.$$

Using Lemma 9, we have the stated bound. ■

Any Hilbert space is $(2, 1)$-strongly convex. As a slightly more exotic example, $\mathbb{R}^d$ equipped with the p-norm is $(2, p - 1)$ strongly-convex for $p \in (1, 2]$.

Appendix B. Proof of the regret bound of ONS in Banach spaces

First, we need some additional facts about self-adjoint operators. These are straight-forward properties in Hilbert spaces, but may be less familiar in Banach spaces so we present them below for completeness.

Proposition 4 *Suppose X and Y are Banach spaces and $T : X \to Y$ is invertible. Then, $T^\star$ is invertible and $(T^{-1})^\star = (T^\star)^{-1}$.*

Proof Let $y^\star \in Y^\star$. Let $x \in X$. Recall that by definition $\langle T^\star(y^\star), x\rangle = \langle y^\star, T(x)\rangle$. Then we have

$$\langle (T^{-1})^\star(T^\star(y^\star)), x\rangle = \langle T^\star(y^\star), T^{-1}(x)\rangle = \langle y^\star, x\rangle,$$

where we used the definition of adjoint twice. Therefore, $(T^{-1})^\star(T^\star(y^\star)) = y^\star$ and so $(T^{-1})^\star = (T^\star)^{-1}$. ■

Proposition 5 *Suppose B is a reflexive Banach space and $T : B \to B^\star$ is such that*

$$T(x) = \sum_{i=1}^{N} \langle b^i, x\rangle b^i$$

for some vectors $b^i \in B^\star$. Then $T^\star = T$.

Proof Let $g, f \in B$. Since B is reflexive, g corresponds to the function $\langle \cdot, g\rangle \in B^{\star\star}$. Now, we compute:

$$T^\star(g)(f) = \langle T(f), g\rangle = \sum_{i=1}^{N} \langle b^i, f\rangle\langle b^i, g\rangle = \langle T(g), f\rangle = T(g)(f)\ .$$

■

Proposition 6 *Suppose $\tau > 0$, B is a d-dimensional real Banach space, $b^1, \dots, b^d$ are a basis for $B^\star$ and $g_1, \dots, g_T$ are elements of $B^\star$. Then, $A : B \to B^\star$ defined by $A(x) = \tau \sum_{i=1}^{d} \langle b^i, x\rangle b^i + \sum_{t=1}^{T} \langle g_t, x\rangle g_t$ is invertible and self-adjoint, and $\langle Ax, x\rangle > 0$ for all $x \neq 0$.*

Proof First, A is self-adjoint by Proposition 5.

Next, we show A is invertible. Suppose otherwise. Then, since B and $B^\star$ are both d-dimensional, A must have a non-trivial kernel element x. Therefore,

$$0 = \langle Ax, x\rangle = \tau \sum_{i=1}^{d} \langle b^i, x\rangle^2 + \sum_{t=1}^{T} \langle g_t, x\rangle^2, \tag{5}$$

so that $\langle b^i, x\rangle = 0$ for all i. Since the b^i form a basis for $B^\star$, this implies $\langle y, x\rangle = 0$ for all $y \in B^\star$, which implies $x = 0$. Therefore, A has no kernel and so must be invertible.

Finally, observe that since (5) holds for any x, we must have $\langle Ax, x\rangle > 0$ if $x \neq 0$. ■

Now we state the ONS algorithm in Banach spaces and prove its regret guarantee:

Algorithm 8 ONS in Banach Spaces

Require: Real Banach space B, convex subset $S \subset B$, initial linear operator $L : B \to B^\star$, $\tau, \beta > 0$

1: **Initialize:** $v_1 = 0 \in S$
2: **for** $t = 1$ **to** T **do**
3: Play v_t
4: Receive $z_t \in B^\star$
5: Set $A_t(x) = \tau L(x) + \sum_{i=1}^t z_i \langle z_i, x\rangle$
6: $v_{t+1} = \Pi_S^{A_t}(v_t - \frac{1}{\beta} A_t^{-1}(z_t))$, where $\Pi_S^{A_t}(x) = \operatorname{argmin}_{y \in S} \langle A_t(y - x), y - x\rangle$
7: **end for**

Theorem 11 *Using the notation of Algorithm 8, suppose $L(x) = \sum_{i=1}^d \langle b^i, x\rangle$ for some basis $b^i \in B^\star$ and that B is d-dimensional. Then for any $\mathring{v} \in S$,*

$$\sum_{t=1}^T \left(\langle z_t, v_t - \mathring{v}\rangle - \frac{\beta}{2} \langle z_t, v_t - \mathring{v}\rangle^2 \right) \le \frac{\beta\tau}{2} \langle L(\mathring{v}), \mathring{v}\rangle + \frac{2}{\beta} \sum_{t=1}^T \langle z_t, A_t^{-1}(z_t)\rangle \,.$$

Proof First, observe by Proposition 6 that A_t is invertible and self-adjoint for all t.

Now, define $x_{t+1} = v_t - \frac{1}{\beta} A_t^{-1}(z_t)$ so that $v_{t+1} = \Pi_S^{A_t}(x_{t+1})$. Then, we have

$$x_{t+1} - \mathring{v} = v_t - \mathring{v} - \frac{1}{\beta} A_t^{-1}(z_t),$$

that implies

$$A_t(x_{t+1} - \mathring{v}) = A_t(v_t - \mathring{v} - \frac{1}{\beta} A_t^{-1}(z_t)) = A_t(v_t - \mathring{v}) - \frac{1}{\beta} z_t,$$

and

$$\begin{aligned}
&\langle A_t(x_{t+1} - \mathring{v}), x_{t+1} - \mathring{v}\rangle \\
&\quad = \langle A_t(v_t - \mathring{v}) - \frac{1}{\beta} z_t, x_{t+1} - \mathring{v}\rangle \\
&\quad = \langle A_t(v_t - \mathring{v}), x_{t+1} - \mathring{v}\rangle - \frac{1}{\beta} \langle z_t, x_{t+1} - \mathring{v}\rangle \\
&\quad = \langle A_t(v_t - \mathring{v}), x_{t+1} - \mathring{v}\rangle - \frac{1}{\beta} \langle z_t, v_t - \mathring{v} - \frac{1}{\beta} A_t^{-1}(z_t)\rangle \\
&\quad = \langle A_t(v_t - \mathring{v}), x_{t+1} - \mathring{v}\rangle - \frac{1}{\beta} \langle z_t, v_t - \mathring{v}\rangle + \frac{1}{\beta^2} \langle z_t, A_t^{-1}(z_t)\rangle \\
&\quad = \langle A_t(v_t - \mathring{v}), v_t - \mathring{v} - \frac{1}{\beta} A_t^{-1}(z_t)\rangle - \frac{1}{\beta} \langle z_t, v_t - \mathring{v}\rangle + \frac{1}{\beta^2} \langle z_t, A_t^{-1}(z_t)\rangle \\
&\quad = \langle A_t(v_t - \mathring{v}), v_t - \mathring{v}\rangle - \frac{1}{\beta} \langle A_t(v_t - \mathring{v}), A_t^{-1}(z_t)\rangle - \frac{1}{\beta} \langle z_t, v_t - \mathring{v}\rangle + \frac{1}{\beta^2} \langle z_t, A_t^{-1}(z_t)\rangle \\
&\quad = \langle A_t(v_t - \mathring{v}), v_t - \mathring{v}\rangle - \frac{2}{\beta} \langle z_t, v_t - \mathring{v}\rangle + \frac{1}{\beta^2} \langle z_t, A_t^{-1}(z_t)\rangle,
\end{aligned}$$

where in the last line we used $\langle A_t(v_t - \mathring{v}), A_t^{-1}(z_t)\rangle = \langle (v_t - \mathring{v}), A_t^\star A_t^{-1}(z_t)\rangle$ and $A_t^\star = A_t$. We now use the Lemma 8 from [12], extended to Banach spaces thanks to the last statement of Proposition 6, to have

$$\langle A_t(x_{t+1} - \mathring{v}), x_{t+1} - \mathring{v}\rangle \geq \langle A_t(v_{t+1} - \mathring{v}), v_{t+1} - \mathring{v}\rangle$$

to have

$$\langle z_t, v_t - \mathring{v}\rangle \leq \frac{\beta}{2}\langle A_t(v_t - \mathring{v}), v_t - \mathring{v}\rangle - \frac{\beta}{2}\langle A_t(v_{t+1} - \mathring{v}), v_{t+1} - \mathring{v}\rangle + \frac{2}{\beta}\langle z_t, A_t^{-1}(z_t)\rangle \,.$$

Summing over $t = 1, \cdots, T$, we have

$$\begin{aligned}
\sum_{t=1}^T \langle z_t, v_t - \mathring{v}\rangle &\leq \frac{\beta}{2}\langle A_1(v_1 - \mathring{v}), v_1 - \mathring{v}\rangle + \frac{\beta}{2}\sum_{t=2}^T \langle A_t(v_t - \mathring{v}) - A_{t-1}(v_t - \mathring{v}), v_t - \mathring{v}\rangle \\
&\quad - \frac{\beta}{2}\langle A_T(v_{T+1} - \mathring{v}), v_{T+1} - \mathring{v}\rangle + \frac{2}{\beta}\sum_{t=1}^T \langle z_t, A_t^{-1}(z_t)\rangle \\
&\leq \frac{\beta}{2}\langle A_1(v_1 - \mathring{v}), v_1 - \mathring{v}\rangle + \frac{\beta}{2}\sum_{t=2}^T \langle z_t\langle z_t, v_t - \mathring{v}\rangle, v_t - \mathring{v}\rangle + \frac{2}{\beta}\sum_{t=1}^T \langle z_t, A_t^{-1}(z_t)\rangle \\
&= \frac{\beta}{2}\langle \tau L(\mathring{v}), \mathring{v}\rangle + \frac{\beta}{2}\sum_{t=1}^T \langle z_t, v_t - \mathring{v}\rangle^2 + \frac{2}{\beta}\sum_{t=1}^T \langle z_t, A_t^{-1}(z_t)\rangle \,.
\end{aligned}$$

■

It remains to choose L properly and analyze the sum $\sum_{t=1}^T \langle z_t, A_t^{-1}(z_t)\rangle$ In order to do this, we introduce the concept of an Auerbach basis (e.g. see [11] Theorem 1.16):

Theorem 12 *Let B be a d-dimensional Banach space. Then there exists a basis of $b_1, \ldots, b_d$ of B and a basis $b^1, \ldots, b^d$ of $B^\star$ such that $\|b_i\| = \|b^i\|_\star = 1$ for all i and $\langle b_i, b^j\rangle = \delta_{ij}$. Any bases (b_i) and (b^i) satisfying these conditions is called an* Auerbach basis.

We will use an Auerbach basis to define L, and also to provide a coordinate system that makes it easier to analyze the sum $\sum_{t=1}^T \langle z_t, A_t^{-1}(z_t)\rangle$.

Theorem 13 *Suppose B is d-dimensional. Let (b_i) and (b^i) be an Auerbach basis for B. Set $L(x) = \sum_{i=1}^d \langle b^i, x\rangle b^i$. Define A_t as in Algorithm 7. Then, for any $\mathring{v} \in S$, the following holds*

$$\frac{\beta\tau}{2}\langle L(\mathring{v}), \mathring{v}\rangle + \frac{2}{\beta}\sum_{t=1}^T \langle z_t, A_t^{-1}(z_t)\rangle \leq \frac{\beta\tau}{2} d\|\mathring{v}\|^2 + \frac{2}{\beta} d \ln\left(\frac{\sum_{t=1}^T \|z_t\|_\star^2}{\tau} + 1\right) \,.$$

Proof First, we show that $\frac{\beta}{2}\langle L(\mathring{v}), \mathring{v}\rangle \leq \frac{\beta d}{2}\|\mathring{v}\|^2$. To see this, observe that for any $x \in B$,

$$\langle L(x), x\rangle = \sum_{i=1}^d \langle b^i, x\rangle^2 \leq \sum_{i=1}^d \|b^i\|_\star^2 \|x\|^2 \leq d\|x\|^2 \,.$$

Now, we characterize the sum part of the bound. The basic idea is to use the Auerbach basis to identify B with $\mathbb{R}^d$ (equivalently, we view $\langle L(x), x\rangle$ as an inner product on B). We use this identification to translate all quantities in B and $B^\star$ to vectors in $\mathbb{R}^d$, and observe that the 2-norm of any g_t in $\mathbb{R}^d$ is at most d. Then we use analysis of the same sum terms in the classical analysis of ONS in $\mathbb{R}^d$ [12] to prove the bound.

We spell these identifications explicitly for clarity. Define a map $F : B \to \mathbb{R}^d$ by

$$F(x) = (\langle b^1, x\rangle, \dots, \langle b^d, x\rangle) .$$

We have an associated map $F^\star : B^\star \to \mathbb{R}^d$ given by

$$F^\star(x^\star) = (\langle x^\star, b_1\rangle, \dots, \langle x^\star, b_d\rangle) .$$

Since $\langle b^i, b_j\rangle = \delta_{ij}$, these maps respect the action of dual vectors in $B^\star$. That is,

$$\langle x, y\rangle = F^\star(x) \cdot F(y) .$$

Further, since each $\|b_i\| = \|b_i\|_\star = 1$, we have

$$\|F(x)\|^2 = \sum_{i=1}^{d} \langle b^i, x\rangle^2 \le d\|x\|^2 .$$

and

$$\|F^\star(x)\|^2 = \sum_{i=1}^{d} \langle x, b_i\rangle^2 \le d\|x\|_\star^2 .$$

where the norm in $\mathbb{R}^d$ is the 2-norm. To make the correspondence notation cleaner, we write $\overline{x} = F(x)$ for $x \in B$ and $\overline{y} = F^\star(y)$ for $y \in B^\star$. $\overline{x}_i$ indicates the ith coordinate of $\overline{x}$.

Given any linear map $M : B \to B^\star$ (which we denote by $M \in \mathcal{L}(B, B^\star)$), there is an associated map $\overline{M} : \mathbb{R}^d \to \mathbb{R}^d$ given by

$$\overline{M} = F^\star M F^{-1} .$$

Further, when written as a matrix, the ijth element of $\overline{M}$ is

$$\overline{M}_{ij} = (F^\star M F^{-1} e_j) \cdot e_i,$$

where e_j represents the jth standard basis element in $\mathbb{R}^d$. A symmetric statement holds for any linear map $B^\star \to B$, in which $\overline{M} = FM(F^\star)^{-1}$.

These maps all commute properly: $\overline{Mx} = \overline{M}\overline{x}$ for any $M \in \mathcal{L}(B, B^\star)$ and $x \in B$, and similarly $\overline{Mx} = \overline{M}\overline{x}$ for any $M \in \mathcal{L}(B^\star, B)$ and $x \in B^\star$. It follows that $\overline{M}^{-1} = \overline{M^{-1}}$ for any M as well.

Now, let's calculate $\overline{L}_{ij}$:

$$\overline{L}_{ij} = (F^\star L F^{-1} e_j) \cdot e_i = \langle L b_j, b_i\rangle = \delta_{ij},$$

so that the matrix $\overline{L}$ is the identity.

Finally, if $M_g : B \to B^\star$ is the map $M_g(x) = \langle g, x\rangle g$, then a simple calculation shows

$$\overline{M_g} = \overline{g}\overline{g}^T .$$

With these details described, recall that we are trying to bound the sum

$$\sum_{t=1}^{T} \langle z_t, A_t^{-1}(z_t)\rangle .$$

We transfer to $\mathbb{R}^d$ coordinates:

$$\sum_{t=1}^{T} \langle z_t, A_t^{-1}(z_t)\rangle = \sum_{t=1}^{T} \overline{z_t} \cdot \overline{A_t}^{-1} \overline{z_t} .$$

We have $\|\overline{z_n}\| \le \sqrt{d}\|z_n\|_\star$ and

$$\overline{A_t} = \tau \overline{L} + \sum_{t=1}^{t} \overline{z_t}\overline{z_t}^T,$$

so that by [12] Lemma 11,

$$\sum_{t=1}^{T} \overline{z_t} \cdot \overline{A_t}^{-1} \overline{z_t} \le \ln \frac{|\overline{A_T}|}{|\overline{A_0}|} \le d \ln \left(\frac{\sum_{t=1}^{T} \|\overline{z_t}\|^2}{d\tau} + 1 \right) \le d \ln \left(\frac{\sum_{t=1}^{T} \|z_t\|_\star^2}{\tau} + 1 \right),$$

where in the second inequality we used the fact that the determinant is maximized when all the eigenvalues are equal to $\frac{\sum_{t=1}^{T} \|\overline{z_t}\|^2}{d}$. ■

For completeness, we also state the regret bound and the setting of the parameters β and τ to obtain a regret bound for exp-concave functions. Note that we use a different settings in Algorithms 1 and 7, tailored to our specific setting.

Theorem 14 *Suppose we run Algorithm 7 on α exp-concave losses. Let D be the diameter of the domain S and $\|\nabla f(x)\|_\star \le Z$ for all the x in S. Then set $\beta = \frac{1}{2}\min\left(\frac{1}{4ZD}, \alpha\right)$ and $\tau = \frac{1}{\beta^2 D^2}$. Then*

$$R_T(\mathring{v}) \le 4d \left(ZD + \frac{1}{\alpha} \right) (1 + \ln(T+1)) .$$

Proof First, observe that classic analysis of α exp-concave functions [12, Lemma 3] shows that for any $x, y \in S$,

$$f(x) \ge f(y) + \langle \nabla f(y), x - y\rangle + \frac{\beta}{2} \langle \nabla f(y), x - y\rangle^2 .$$

(Note that although the original proof is stated in $\mathbb{R}^d$, the exact same argument applies in a Banach space)

Therefore, by Theorems 11 and 13, we have

$$R_T(u) \le \frac{\beta\tau}{2} d\|u\|^2 + \frac{2}{\beta} d \ln(Z^2 T/\tau + 1) .$$

Substitute our values for β and τ to conclude

$$R_T(u) \leq \frac{d}{2\beta}\left(1+\ln(Z^2T\beta^2D^2+1)\right) \leq 4d\left(ZD+\frac{1}{\alpha}\right)(1+\ln(T+1)),$$

where in the last line we used $\frac{1}{\beta} \leq 8(ZD+1/\alpha)$. ■

Appendix C. Proofs of Theorems 1 and 8

In order to prove Theorem 1 and 8, we first need some technical lemmas. In particular, first we show in Lemma 17 that ONS gives us a logarithmic regret against the functions $\ell_t(\beta) = \ln(1+\langle g_t, \beta\rangle)$. Then, we will link the wealth to the regret with respect to an arbitrary unitary vector thanks to Theorem 21.

Lemma 15 *For $-1 < x \leq 2$, we have*

$$\ln(1+x) \leq x - \frac{2-\ln(3)}{4}x^2 \,.$$

Lemma 16 *Define $\ell_t(v) = -\ln(1-\langle g_t, v\rangle)$. Let $\|\mathring{v}\|, \|v\| \leq \frac{1}{2}$ and $\|g_t\|_\star \leq 1$. Then*

$$\ell_t(v) - \ell_t(\mathring{v}) \leq \langle \nabla\ell_t(v), v-\mathring{v}\rangle - \frac{2-\ln(3)}{2}\frac{1}{2}\langle \nabla\ell_t(v), v-\mathring{v}\rangle^2 \,.$$

Proof We have

$$\ln(1-\langle g_t,\mathring{v}\rangle) = \ln(1-\langle g_t,v\rangle + \langle g_t, v-\mathring{v}\rangle) = \ln(1-\langle g_t,v\rangle) + \ln\left(1+\frac{\langle g_t, v-\mathring{v}\rangle}{1-\langle g_t,v\rangle}\right) \,.$$

Now, observe that since $1-\langle g_t,\mathring{v}\rangle \geq 0$ and $1-\langle g_t,v\rangle \geq 0$, $1+\frac{\langle g_t,v-\mathring{v}\rangle}{1-\langle g_t,v\rangle} \geq 0$ as well so that $\frac{\langle g_t,v-\mathring{v}\rangle}{1-\langle g_t,v\rangle} \geq -1$. Further, since $\|\mathring{v}-v\| \leq 1$ and $1-\langle g_t,v\rangle \geq 1/2$, $\frac{\langle g_t,v-\mathring{v}\rangle}{1-\langle g_t,v\rangle} \leq 2$. Therefore, by Lemma 15 we have

$$\ln(1-\langle g_t,\mathring{v}\rangle) \leq \ln(1-\langle g_t,v\rangle) + \frac{\langle g_t, v-\mathring{v}\rangle}{1-\langle g_t,v\rangle} - \frac{2-\ln(3)}{4}\frac{\langle g_t, v-\mathring{v}\rangle^2}{(1-\langle g_t,v\rangle)^2} \,.$$

Using the fact that $\nabla\ell_t(v) = \frac{g_t}{1-\langle g_t,v\rangle}$ finishes the proof. ■

Lemma 17 *Define $S = \{v \in B : \|v\| \leq \frac{1}{2}\}$ and $\ell_t(v) : S \to \mathbb{R}$ as $\ell_t(v) = -\ln(1-\langle g_t, v\rangle)$, where $\|g_t\|_\star \leq 1$. If we run ONS in Algorithm 7 with $\beta = \frac{2-\ln(3)}{2}$, $\tau = 1$, and $S = \{v : \|v\| \leq \frac{1}{2}\}$, then*

$$\sum_{t=1}^T \ell_t(v_t) - \ell_t(\mathring{v}) \leq d\left(\frac{1}{17} + 4.5\ln\left(1+4\sum_{t=1}^T \|g_t\|_\star^2\right)\right) \,.$$

Proof From Lemma 16, we have

$$\sum_{t=1}^{T} \ell_t(v_t) - \ell_t(\mathring{v}) \leq \sum_{t=1}^{T} \left(\langle \nabla \ell_t(v_t), v_t - \mathring{v} \rangle - \frac{\beta}{2} \langle \nabla \ell_t(v_t), v_t - \mathring{v} \rangle^2 \right) .$$

So, using Lemma 11 we have

$$\sum_{t=1}^{T} \left(\langle \nabla \ell_t(v_t), v_t - \mathring{v} \rangle - \frac{\beta}{2} \langle \nabla \ell_t(v_t), v_t - \mathring{v} \rangle^2 \right) \leq \frac{\beta}{2} \langle L(\mathring{v}), \mathring{v} \rangle + \frac{2}{\beta} \sum_{t=1}^{T} \langle z_t, A_t^{-1}(z_t) \rangle,$$

where $z_t = \nabla \ell_t(v_t)$. Now, use Theorem 13 so that

$$\frac{\beta}{2} \langle L(\mathring{v}), \mathring{v} \rangle + \frac{2}{\beta} \sum_{t=1}^{T} \langle z_t, A_t^{-1}(z_t) \rangle \leq \frac{d\beta}{8} + \frac{2d}{\beta} \ln \left(1 + \sum_{t=1}^{T} \|z_t\|_\star^2 \right),$$

where we have used $\|\mathring{v}\| \leq 1/2$. Then observe that $\|z_t\|_\star^2 = \frac{\|g_t\|_\star^2}{(1+\langle g_t, \beta_t \rangle)^2} \leq 4\|g_t\|_\star^2$ so that $\ln(1 + \sum_{t=1}^{T} \|z_t\|_\star^2) \leq \ln(1 + 4\sum_{t=1}^{T} \|g_t\|_\star^2)$. Finally, substitute the specified value of β and numerically evaluate to conclude the bound. ■

Now, we collect some Fenchel conjugate calculations that allow us to convert our wealth lower-bounds into regret upper-bounds:

Lemma 18 *Let $f(x) = a \exp(b|x|)$, where $a, b > 0$. Then*

$$f^\star(\theta) = \begin{cases} \frac{|\theta|}{b} \left(\ln \frac{|\theta|}{ab} - 1 \right), & \frac{|\theta|}{ab} > 1 \\ -a, & \text{otherwise.} \end{cases} \leq \frac{|\theta|}{b} \left(\ln \frac{|\theta|}{ab} - 1 \right) .$$

Lemma 19 *Let $f(x) = a \exp(b \frac{x^2}{|x|+c})$, where $a, b > 0$ and $c \geq 0$. Then*

$$f^\star(\theta) \leq |\theta| \max \left(\frac{2}{b} \left(\ln \frac{2|\theta|}{ab} - 1 \right), \sqrt{\frac{c}{b} \ln \left(\frac{c\theta^2}{a^2 b} + 1 \right)} - a \right) .$$

Proof By definition we have

$$f^\star(\theta) = \sup_x \ \theta x - f(x) .$$

It is easy to see that the sup cannot attained at infinity, hence we can safely assume that it is attained at $x^\star \in \mathbb{R}$. We now do a case analysis, based on $x^\star$.

Case $|x^\star| \leq c$**.** In this case, we have that $f(x^\star) \geq a \exp(b \frac{x^2}{2c})$, so

$$\begin{aligned} f^\star(\theta) = \theta x^\star - f(x^\star) &\leq \theta x^\star - a \exp \left(b \frac{(x^\star)^2}{2c} \right) \\ &\leq \sup_x \ \theta x - a \exp \left(b \frac{x^2}{2c} \right) \leq |\theta| \sqrt{\frac{c}{b} \ln \left(\frac{c\theta^2}{a^2 b} + 1 \right)} - a, \end{aligned}$$

where the last inequality is from Lemma 18 in [23].

Case $|x^\star| > c$**.** In this case, we have that $f(x^\star) \geq a \exp\left(b\frac{(x^\star)^2}{2|x^\star|}\right) = a\exp\left(\frac{b}{2}|x^\star|\right)$, so

$$\begin{aligned} f^\star(\theta) = \theta x^\star - f(x^\star) &\leq \theta x^\star - a\exp\left(\frac{b}{2}|x^\star|\right) \\ &\leq \sup_x \ \theta x - a\exp\left(\frac{b}{2}|x|\right) \leq \frac{2|\theta|}{b}\left(\ln\frac{2|\theta|}{ab} - 1\right), \end{aligned}$$

where the last inequality is from Lemma 18.

Considering the max over the two cases gives the stated bound. ■

Theorem 20 *Let u be an arbitrary unit vector and $\|g_t\|_\star \leq 1$ for $t = 1, \cdots, T$. Then*

$$\sup_{\|v\|\leq\frac{1}{2}} \sum_{t=1}^T \ln(1 - \langle g_t, v\rangle) \geq \frac{1}{4}\frac{\langle\sum_{t=1}^T g_t, u\rangle^2}{\sum_{t=1}^T\langle g_t, u\rangle^2 + \left|\left\langle\sum_{t=1}^T g_t, u\right\rangle\right|}.$$

Proof Recall that $\ln(1+x) \geq x - x^2$ for $|x| \leq 1/2$. Then, we compute

$$\begin{aligned} \sup_{\|v\|\leq 1/2} \sum_{t=1}^T \ln(1 - \langle g_t, v\rangle) &\geq \sup_{\|v\|\leq 1/2} \sum_{t=1}^T \left(-\langle g_t, v\rangle - \langle g_t, v\rangle^2\right) \\ &= \sup_{\|v\|\leq 1/2} -\left\langle\sum_{t=1}^T g_t, v\right\rangle - \sum_{t=1}^T\langle g_t, v\rangle^2. \end{aligned}$$

Choose $v = \frac{u}{2}\frac{\left\langle\sum_{t=1}^T g_t, u\right\rangle}{\sum_{t=1}^T\langle g_t, u\rangle^2 + \left|\left\langle\sum_{t=1}^T g_t, u\right\rangle\right|}$. Then, clearly $\|v\| \leq \frac{1}{2}$. Thus, we have

$$\begin{aligned} \sup_{\|v\|\leq 1/2} \sum_{t=1}^T \ln(1 + \langle g_t, v\rangle) &\geq \sup_{\|v\|\leq 1/2} -\left\langle\sum_{t=1}^T g_t, v\right\rangle - \sum_{t=1}^T\langle g_t, v\rangle^2 \\ &\geq \frac{1}{2}\frac{\langle\sum_{t=1}^T g_t, u\rangle^2}{\sum_{t=1}^T\langle g_t, u\rangle^2 + \left|\left\langle\sum_{t=1}^T g_t, u\right\rangle\right|} - \frac{\langle\sum_{t=1}^T g_t, u\rangle^2}{4\left(\sum_{t=1}^T\langle g_t, u\rangle^2 + \left|\left\langle\sum_{t=1}^T g_t, u\right\rangle\right|\right)^2}\sum_{t=1}^T\langle g_t, u\rangle^2 \\ &\geq \frac{1}{4}\frac{\langle\sum_{t=1}^T g_t, u\rangle^2}{\sum_{t=1}^T\langle g_t, u\rangle^2 + \left|\left\langle\sum_{t=1}^T g_t, u\right\rangle\right|}. \end{aligned}$$

■

Lemma 21 *Let u be an arbitrary unit vector in B and $t > 0$. Then, using the Algorithm 7, we have*

$$R_T(tu) \leq \epsilon + t\max\left[\frac{d}{2} - 8 + 8\ln\frac{8t\left(4\sum_{t=1}^T\|g_t\|_\star^2 + 1\right)^{4.5d}}{\epsilon},\right.$$

$$\left.2\sqrt{\sum_{t=1}^T\langle g_t, u\rangle^2\ln\left(\frac{5t^2}{\epsilon^2}\exp\left(\frac{d}{17}\right)\left(4\sum_{t=1}^T\|g_t\|^2 + 1\right)^{9d+1} + 1\right)}\right].$$

Proof Let's compute a bound on our wealth, Wealth_T. We have that

$$\text{Wealth}_t = \text{Wealth}_{t-1} - \langle g_t, w_t\rangle = \text{Wealth}_{t-1}(1 - \langle g_t, v_t\rangle) = \epsilon\prod_{t=1}^{T}(1 - \langle g_t, v_t\rangle),$$

and taking the logarithm we have

$$\ln \text{Wealth}_t = \ln\epsilon + \sum_{t=1}^{T}\ln(1 - \langle g_t, v_t\rangle)\ .$$

Hence, using Lemma 17, we have

$$\ln \text{Wealth}_t \geq \ln\epsilon + \max_{\|v\|\leq\frac{1}{2}}\sum_{t=1}^{T}\ln(1 + \langle g_t, v\rangle) - d\left(\frac{1}{17} + 4.5\ln\left(1 + \sum_{t=1}^{T}4\|g_t\|_\star^2\right)\right)\ .$$

Using Theorem 20, we have

$$\text{Wealth}_T \geq \frac{\epsilon}{\exp\left[d\left(\frac{1}{17} + 4.5\ln\left(1 + 4\sum_{t=1}^{T}\|g_t\|_\star^2\right)\right)\right]}\exp\left[\frac{1}{4}\frac{\langle\sum_{t=1}^{T}g_t, u\rangle^2}{\sum_{t=1}^{T}\langle g_t, u\rangle^2 + \left|\left\langle\sum_{t=1}^{T}g_t, u\right\rangle\right|}\right]\ .$$

Defining

$$f(x) = \frac{\epsilon}{\exp\left[d\left(\frac{1}{17} + 4.5\ln\left(1 + 4\sum_{t=1}^{T}\|g_t\|_\star^2\right)\right)\right]}\exp\left[\frac{1}{4}\frac{x^2}{\sum_{t=1}^{T}\langle g_t, u\rangle^2 + |x|}\right],$$

we have

$$\begin{aligned}
R_T(tu) &= \epsilon - \text{Wealth}_T - t\left\langle\sum_{t=1}^{T}g_t, u\right\rangle\\
&\leq \epsilon - t\left\langle\sum_{t=1}^{T}g_t, u\right\rangle - f\left(\left\langle\sum_{t=1}^{T}g_t, u\right\rangle\right)\\
&\leq \epsilon + f^\star(-t)\\
&\leq \epsilon + t\max\left[8\left(\ln\frac{8t}{\epsilon} + \frac{d}{17} + 4.5d\ln\left(4\sum_{t=1}^{T}\|g_t\|_\star^2 + 1\right) - 1\right),\right.\\
&\qquad\left.\sqrt{4\sum_{t=1}^{T}\langle g_t, u\rangle^2\ln\left(\frac{5t^2}{\epsilon^2}\exp\left(\frac{d}{17}\right)\left(4\sum_{t=1}^{T}\|g_t\|^2 + 1\right)^{9d}\sum_{t=1}^{T}\langle g_t, u\rangle^2 + 1\right)}\right]\\
&\leq \epsilon + t\max\left[\frac{d}{2} - 8 + 8\ln\frac{8t\left(4\sum_{t=1}^{T}\|g_t\|_\star^2 + 1\right)^{4.5d}}{\epsilon},\right.\\
&\qquad\left.2\sqrt{\sum_{t=1}^{T}\langle g_t, u\rangle^2\ln\left(\frac{5t^2}{\epsilon^2}\exp\left(\frac{d}{17}\right)\left(4\sum_{t=1}^{T}\|g_t\|^2 + 1\right)^{9d+1} + 1\right)}\right],
\end{aligned}$$

where we have used the calculation of Fenchel conjugate of f from Lemma 19. Then observe that $\exp(d/17) \leq \exp((9d+1)/153) \leq 2^{9d+1}$ to conclude:

$$R_T(tu) \leq \epsilon + t \max \left[\frac{d}{2} - 8 + 8 \ln \frac{8t \left(4 \sum_{t=1}^T \|g_t\|_\star^2 + 1\right)^{4.5d}}{\epsilon}, \right.$$
$$\left. 2 \sqrt{\sum_{t=1}^T \langle g_t, u \rangle^2 \ln \left(\frac{5t^2}{\epsilon^2} \left(8 \sum_{t=1}^T \|g_t\|^2 + 2 \right)^{9d+1} + 1 \right)} \right] .$$

■

Proof [Proof of Theorem 8] Given some $\mathring{w}$, set $u = \frac{\mathring{w}}{\|\mathring{w}\|}$ and $t = \|\mathring{w}\|$. Then observe that $t^2 \sum_{t=1}^T \langle g_t, u \rangle^2 = \sum_{t=1}^T \langle g_t, \mathring{w} \rangle^2$ and apply the previous Lemma 21 to conclude the desired result. ■

Appendix D. Proof of Proposition 1 and Theorem 4

We restate Proposition 1 below:

Proposition 1 *S_W is convex and 1-Lipschitz for any closed convex set W in a reflexive Banach space B.*

Proof Let $x, y \in B$, $t \in [0,1]$, $x' \in \Pi_W(x)$, and $y' \in \Pi_W(y)$. Then

$$\begin{aligned} S_W(tx + (1-t)y) &= \min_{d \in W} \|tx + (1-t)y - d\| \leq \|tx + (1-t)y - tx' - (1-t)y'\| \\ &= \|t(x - x') + (1-t)(y - y')\| \leq t\|x - x'\| + (1-t)\|y - y'\| \\ &= tS_W(x) + (1-t)S_W(y) . \end{aligned}$$

For the Lipschitzness, let $x \in B$ and $x' \in \Pi_W(x)$, and observe that

$$S_W(x + \delta) = \inf_{d \in W} \|x + \delta - d\| \leq \|x + \delta - x'\| \leq S_W(x) + \|\delta\| .$$

Similarly, let $x \in B$, δ such that $x + \delta \in B$ and $x' \in \Pi_W(x + \delta)$, then

$$S_W(x) = \min_{d \in W} \|x - d\| \leq \|x + \delta - \delta - x'\| \leq S_W(x + \delta) + \|\delta\| .$$

So that $|S_W(x) - S_W(x + \delta)| \leq \|\delta\|$. ■

Now we restate and prove Theorem 4:

Theorem 4 *Let B be a reflexive Banach space such that for every $0 \neq b \in B$, there is a unique dual vector $b^\star$ such that $\|b^\star\|_\star = 1$ and $\langle b^\star, b \rangle = \|b\|$. Let $W \subset B$ a closed convex set. Given $x \in B$ and $x \notin W$, let $p \in \Pi_W(x)$. Then $\{(x - p)^\star\} = \partial S_W(x)$.*

Proof Let $x' = \frac{x+p}{2}$. Then clearly $S_W(x') \leq \|x' - p\| = \frac{\|x-p\|}{2} = S_W(x) - \|x - x'\|$. Since S_W is 1-Lipschitz, $S_W(x') \geq S_W(x) - \|x - x'\|$ and so $S_W(x') = S_W(x) - \|x - x'\|$.

Suppose $g \in \partial S_W(x)$. Then $\langle g, x' - x\rangle + S_W(x) \leq S_W(x') = S_W(x) - \|x - x'\|$. Therefore, $\langle g, x' - x\rangle \leq -\|x - x'\|$. Since $\|g\|_\star \leq 1$, we must have $\|g\|_\star = 1$ and $\langle g, x - p\rangle = \|x - p\|$. By assumption, this uniquely specifies the vector $(x-p)^\star$. Since ∂S_W is not the empty set, $\{(x-p)^\star\} = \partial S_W(x)$. ■

Appendix E. Computing S_W for multi-scale experts

In this section we show how to compute $\Pi_W(x)$ and a subgradient of $S_W(x)$ in Algorithm 5. First we tackle $\Pi_W(x)$. Without loss of generality, assume the c_i are ordered so that $c_1 \geq c_2 \geq \cdots \geq c_N$. We also consider $W_k = \{x : x_i \geq 0 \text{ for all } i \text{ and } \sum_{i=1}^N x_i/c_i = k\}$ instead of $W = W_1$. Obviously we are particularly interested in the case $k = 1$, but working in this mild generality allows us to more easily state an algorithm for computing $\Pi_W(x)$ in a recursive manner.

Proposition 7 *Let $N > 1$ and $W_k = \{x : x_i \geq 0$ for all i and $\sum_{i=1}^N x_i/c_i = k\}$, and let $S_{W_k}(x) = \inf_{y \in W_k} \|x - y\|_1$. Suppose the c_i are ordered so that $c_1 \geq c_2 \geq \cdots \geq c_N$. Then for any $x = (x_1, \ldots, x_n)$, there exists a $y = (y_1, \ldots, y_n) \in \Pi_{W_k}(x)$ such that*

$$y_1 = \begin{cases} 0, & x_1 < 0 \\ x_1, & x_1 \in [0, kc_1] \\ kc_1, & x_1 > kc_1 \end{cases}$$

Proof First, suppose $N = 1$. Then clearly there is only one element of W_k and so the choice of $\Pi_{W_k}(x)$ is forced. So now assume $N > 1$.

Let $(y_1, \ldots, y_N) \in \Pi_{W_k}(x_1, \ldots, x_N)$ be such that $|y_1 - x_1|$ is as small as possible (such a point exists because W_k is compact).

We consider three cases: either $x_1 > kc_1$, $x_1 < 0$ or $x_1 \in [0, kc_1]$.

Case 1: $x > kc_1$. Suppose $y_1 < kc_1$. Let i be the largest index such that $y_i \neq 0$. $i \neq 1$ since $y_1/c_1 < k$. Choose $0 < \epsilon < \min(y_i \frac{c_1}{c_i}, kc_1 - y_1)$. Then let y' be such that $y'_1 = y_1 + \epsilon$, $y'_i = y_i - \epsilon \frac{c_i}{c_1}$ and $y'_j = y_j$ otherwise. Then by definition of ϵ, $y'_i \geq 0$ and $y'_1 \leq kc_1$. Further, $\sum_{j=1}^N y'_j/c_j = \epsilon/c_1 - \frac{c_i}{c_1}\epsilon/c_i + \sum_{j=1}^N y_j/c_j = k$ so that $y' \in W_k$. However, since $x_1 > kc_1$, $\|y' - x\|_1 \leq \|y - x\|_1 - \epsilon + \epsilon\frac{c_i}{c_1} \leq \|y - x\|_1$. Therefore, $y' \in \Pi_{W_k}(x)$, but $|y'_1 - x_1| < |y_1 - x_1|$, contradicting our choice of y_1. Therefore, $y_1 = kc_1$.

Case 2: $x < 0$. This case is very similar to the previous case. Suppose $y_1 > 0$. Let i be the largest index such that $y_i \neq kc_i$. $i \neq 1$ since otherwise $\sum_{j=1}^N y_j/c_j > \sum_{j=2}^N k = k(N-1) \geq k$, which is not possible. Choose $0 < \epsilon < \min(y_1, c_1(kc_i - y_i)/c_i)$. Set y' such that $y'_1 = y_1 - \epsilon$, $y'_i = y_i + \epsilon\frac{c_i}{c_1}$. Then, again we have $y' \in W_k$ and $\|y' - x\|_1 \leq \|y - x_1\|_1 - \epsilon + \epsilon\frac{c_i}{c_1} \leq \|y - x\|_1$ so that $y' \in \Pi_{W_k}(x)$, but $|y'_1 - x_1| < |y_1 - x_1|$. Therefore, we cannot have $y_1 > 0$ and so $y_1 = 0$.

Case 3: $x \in [0, kc_1]$. Suppose $y_1 < x_1 \leq kc_1$. Then by the same the argument as for Case 1, there is some $i > 1$ such that for any $0 < \epsilon < \min(y_i \frac{c_1}{c_i}, x_1 - y_1)$, we can construct y' with $y' \in \Pi_{W_k}(x)$ and $|y'_1 - x_1| < |y_1 - x_1|$. Therefore, $y_1 \geq x_1$.

Similarly, if $y_1 > x_1$, then by the same argument as for Case 2, there is some $i > 1$ such that for any $0 < \epsilon < \min(y_1 - x_1, c_1(kc_i - y_i)/c_i)$, we again construct y' with $y' \in \Pi_{W_k}(x)$ and $|y'_1 - x_1| < |y_1 - x_1|$. Therefore, $y_1 = x_1$. ■

This result suggests an explicit algorithm for choosing $y \in \Pi_W(x) = \Pi_{W_1}(x)$. Using the Proposition we can pick y_1 such that there is a $y \in \Pi_{W_1}(x)$ with first coordinate y_1. If $y \in \Pi_{W_k}(x)$ has first coordinate y_1, then if $W_k^2 = \{(y_2, \dots, y_n) : y_i \geq 0 \text{ for all } i \text{ and } \sum_{i=2}^N y_i/c_i = k\}$, then $(y_2, \dots, y_N) \in \Pi_{W_{k-y_1/c_1}^2}(x_2, \dots, x_N)$. Therefore, we can use a greedy algorithm to choose each y_i in increasing order of i and obtain a point $y \in \Pi_{W_k}(x)$ in $O(N)$ time. This procedure is formalized in Algorithm 9.

Algorithm 9 Computing $\Pi_W(x)$

Require: $(x_1, \dots, x_N) \in \mathbb{R}^N$
Initialize: $k_1 = 1$, $i = 1$
for $i = 1$ **to** N **do**
  if $i = N$ **then**
    Set $y_i = k_i c_i$
  else
    if $x_i \leq 0$ **then**
      Set $y_i = 0$
    end if
    if $x_i > k_i c_i$ **then**
      Set $y_i = k_i c_i$
    end if
    if $x_i \in (0, k_i c_i]$ **then**
      Set $y_i = x_i$
    end if
    Set $k_{i+1} = k_i - y_i/c_i$
  end if
end for
return $(y_1, \dots, y_N)$

E.1. Computing a subgradient of S_W for multi-scale experts

Unfortunately, $\|\cdot\|_1$ does not satisfy the hypotheses of Theorem 4 and so we need to do a little more work to compute a subgradient.

Proposition 8 *Let $(y_1, \dots, y_n)$ be the output of Algorithm 9 on input $x = (x_1, \dots, x_N)$. Then if $i = N$, $\frac{\partial S_W(x)}{\partial x_i} = \operatorname{sign}(x_N - y_N)$. Let M be the smallest index such that $y_M = k_M c_M$, where k_i is defined in Algorithm 9. There exists a subgradient $g \in \partial S_W(x)$ such that*

$$g_i = \begin{cases} -1, & x_i \leq 0 \\ 1, & x_i > k_i c_i \\ \operatorname{sign}(x_M - y_M)\frac{c_M}{c_i}, & x_i \in (0, k_i c_i],\ x_M \neq k_M c_M \\ \frac{c_M}{c_i}, & x_i \in (0, k_i c_i],\ x_M = k_M c_M \end{cases}$$

Proof We start with a few reductions. First, we show that by a small perturbation argument we can assume $x_M \neq k_M c_M$. Next, we show that it suffices to prove that S_W is linear on a small L_∞

ball near x. Then we go about proving the Proposition for that L_∞ ball, which is the meat of the argument.

Before we start the perturbation argument, we need a couple observations about M. First, observe that $k_i = y_i = 0$ for all $i > M$.

Next, we show that either have $M = N$, or $x_M \geq k_M c_M$. If $M \neq N$, then by inspection of the Algorithm 9, we must have $x_M \leq 0$ and $k_M = 0$ or $x_M \geq k_M c_M$. If $k_M = 0$, then we have $0 = k_M = k_{M-1} - \frac{y_{M-1}}{c_{M-1}}$. This implies $k_{M-1}c_{M-1} = y_{M-1}$, which contradicts our choice of M as the smallest index with $y_M = k_M c_M$. Therefore, we must have $x_M \geq k_M c_M$. Therefore, we must have $M = N$, or $x_M \geq k_M c_M$.

Now, we show that we may assume $x_M \neq k_M c_M$. Let $\delta > 0$. If $x_M \neq k_M c_M$, set $x_\delta = x$. Otherwise, set $x_\delta = x + \delta e_M$. By inspecting Algorithm 9, we observe that the output on x_δ is unchanged from the output on x, and M is still the smallest index such that $y_i = k_i c_i$.

We claim that it suffices to prove $g \in \partial S_W(x_\delta)$ for all δ rather than $g \in \partial S_W(x)$. To see this, observe that by 1-Lipschitzness, $|S_W(x_\delta) - S_W(x)| \leq \delta$, so that if $g \in \partial S_W(x_\delta)$, then for any w,

$$S_W(w) \geq S_W(x_\delta) + \langle g, w - x_\delta \rangle \geq S_W(x) + \langle g, w - x \rangle - 2\delta \; .$$

By taking $\delta \to 0$, we see that g must be a subgradient of S_W at x if $g \in \partial S_W(x_\delta)$ for all δ. This implies that if we prove the Proposition for any x_δ, which has $x_M \neq k_M c_M$, we have proved the proposition for x.

Following this perturbation argument, for the rest of the proof we consider only the case $x_M \neq k_M c_M$.

Now, we claim that to show the Proposition, it suffices to exhibit a closed L_∞ ball B such that x is on the boundary of B and for $z \in B$, $S_W(z) = \langle g, z \rangle + F$ for some constant F. To see this, first suppose that we have such a B. Then observe that g is the derivative, and therefore a subgradient, of S_W for any point in the interior of B. Let z be in the interior of B and let w be an arbitrary point in $\mathbb{R}^N$. Then since g is a subgradient at z, we have $S_W(w) \geq S_W(z) + \langle g, w - z \rangle$. Further, since x is on the boundary of B (and therefore in B), $S_W(x) = S_W(z) + \langle g, x - z \rangle$. Putting these identities together:

$$\begin{aligned} S_W(w) &\geq S_W(z) + \langle g, w - z \rangle \\ &= S_W(z) + \langle g, x - z \rangle + \langle g, w - x \rangle \\ &= S_W(x) + \langle g, w - x \rangle \; . \end{aligned}$$

Therefore, g is a subgradient of S_W at x.

Next, we turn to identifying the particular L_∞ ball we will work with. Let

$$\begin{aligned} q &= \frac{1}{2} \min_{x_i > 0} x_i, \\ d &= \frac{1}{2} \min_{j | x_j \neq k_j c_j} \min(1/c_1, 1) |x_j - c_j k_j|, \\ h &= \min(q, d) \min(c_N, 1)/N \; . \end{aligned}$$

Consider the L_∞ ball given by

$$B = \{x + (\epsilon_1, \ldots, \epsilon_N) | \; \epsilon_j \in [-h, 0]\} \; .$$

Clearly, x is on the boundary of B. Now, we proceed to show that S_W is linear on the interior of B, which will prove the Proposition by the above discussion.

Let $x' = x + \epsilon$ be an element of B. We will compute $S_W(x')$ by computing the output y' of running Algorithm 9 on x'. We will also refer to the internally generated variables k_i as k'_i to distinguish between the ks generated when computing y versus when computing y'. The overall strategy is to show that all of the conditional branches in Algorithm 9 will evaluate to the same branch on x as on x'.

Specifically we show the following claim by induction:

Claim 9 *for any $i < M$:*

$$y'_i = \begin{cases} 0 & x_i \le 0 \\ x'_i & x_i \in (0, k_i c_i] \end{cases},$$
$$k'_{i+1} = k_{i+1} + \sum_{j \le i,\ x_j \in (0, k_j c_j]} -\epsilon_j / c_j,$$
$$k_{i+1} \le k'_{i+1} \le k_{i+1} + d\frac{i}{2N},$$
$$|y'_i - x'_i| = \begin{cases} |y_i - x_i| - \epsilon_i & x_i \le 0 \\ |y_i - x_i| & x_i \in (0, k_i c_i] \end{cases}.$$

For $i = M$,

$$y'_i = k'_i c_i,$$
$$k'_{i+1} = 0,$$
$$|y'_i - x'_i| = |y_i - x_i| + \operatorname{sign}(x_i - y_i)\epsilon_M + \sum_{j < M |\ x_j \in (0, k_j c_j]} c_M \epsilon_j / c_j\ .$$

And for $i > M$:

$$y'_i = 0,$$
$$k'_{i+1} = 0,$$
$$|y'_i - x'_i| = \begin{cases} |y_i - x_i| - \epsilon_i & x_i \le 0 \\ |y_i - x_i| + \epsilon_i & x_i > 0 \end{cases}.$$

First we do the base case. Observe that $k'_1 = k_1$. Then we consider three cases, either $x_1 \le 0$, $x_1 \in (0, k_1 c_1]$, or $x_1 > k_1 c_1$. These cases correspond to $y_1 = 0$, $y_1 = x_1$, or $y_1 = k_1 c_1$.

Case 1 ($x_1 \le 0$): Since $\epsilon_1 \le 0$, we have $x'_1 = x_1 + \epsilon_1 \le 0$. Therefore, by inspecting the condition blocks in Algorithm 9, $y'_1 = y_1 = 0$ and $k'_2 = k_2$.

Case 2 ($x_1 \in (0, k_1, c_1]$): Since $x_1 > 0$, we have $|\epsilon_1| \le q \le x_1/2$. Therefore, $x'_1 > 0$. Since $\epsilon_1 \le 0$, $x'_1 \le x_1 \le k_1 c_1 = k'_1 c_1$ so that $x'_1 \in (0, k'_1 c_1]$. This implies $y'_1 = x'_1$ and

$$\begin{aligned} k'_2 &= k'_1 - \frac{x'_1}{c_1} \\ &= k_1 - \frac{x_1 + \epsilon_1}{c_1} \\ &= k_2 - \frac{\epsilon_1}{c_1}\ . \end{aligned}$$

Case 3 ($x_1 > k_1 c_1$)**:** In this last case, observe that $|\epsilon_1| < d \le (x_1 - k_1 c_1)/2$ so that $x_1 \ge x_1' > k_1 c_1 = k_1' c_1$. This implies $y_1' = k_1' c_1 = k_1 c_1$ and $k_2' = 0$.

The values for $|y_1' - x_1'|$ can also be checked via the casework. First, suppose $1 = M$. Then we must have $x_1 > k_1 c_1$ (because we assume $x_M \neq k_M c_M$ by our perturbation argument). Therefore, $y_1 = y_1' = k_1 c_1$ and the base case is true.

When $1 < M$, then we consider the cases $x_1 \le 0$ and $x_1 \in (0, k_1 c_1]$. The case $x_1 > k_1 c_1$ does not occur because $1 < M$. When $x_1 \le 0$, then by the above casework we must have $x_1' \le 0$ and $y_1' = y_1 = 0$. Therefore,

$$|y_1' - x_1'| = |x_1'| = |x_1| + |\epsilon_1| = |y_1 - x_1| - \epsilon_1,$$

where we have used $\epsilon_1 \le 0$ to conclude $|x_1'| = |x_1| + |\epsilon_1|$.

When $x_1 \in (0, k_1 c_1]$, we have $y_1 = x_1$, and by the above casework we have and $y_1' = x_1'$. Thus $|y_1' - x_1'| = 0 = |y_1 - x_1|$. This concludes the base case of the induction.

Now, we move on to the inductive step. Suppose the claim holds for all $j < i$. To show the claim also holds for i, we consider the three cases $i < M$, $i = M$ and $i > M$ separately:

Case 1 ($i < M$)**:** We must consider two sub-cases, either $x_i \le 0$, or $x_i \in (0, k_i c_i]$. The case $x_i > k_i c_i$ does not occur because $i < M$.

Case 1a ($x_i \le 0$)**:** In this case, we have $y_i = 0$ and $k_{i+1} = k_i$. By definition, $\epsilon_i \le 0$ so that $x_i' \le 0$. Then by inspection of Algorithm 9, $y_i' = 0 = y_i$ so that $k_{i+1}' = k_i'$. By the induction assumption, this implies

$$k_{i+1}' = k_i' = k_i + \sum_{j<i,\ x_j \in (0, k_j c_j]} -\epsilon_j / c_j = k_{i+1} + \sum_{j \le i,\ x_j \in (0, k_j c_j]} -\epsilon_j / c_j \ .$$

Also, $k_{i+1}' = k_i' \ge k_i = k_{i+1}$ and also

$$|k_{i+1}' - k_{i+1}| = |k_i' - k_i| \le d \frac{i-1}{N} \le d \frac{i}{N} \ .$$

Finally, since $y_i' = 0 = y_i$ and $x_i, x_i' \le 0$, we have

$$|y_i' - x_i'| = |x_i'| = -x_i' = -x_i - \epsilon_i = |x_i| - \epsilon_i = |y_i - x_i| - \epsilon_i \ .$$

Thus all parts of the claim continue to hold.

Case 1b ($x_i \in (0, k_i c_i]$)**:** In this case we show that $x_i' \in (0, k_i', c_i]$. Observe that $y_i = x_i$ and $k_{i+1} = k_i - x_i / c_i$. By definition again, $\epsilon_i \le 0$, and also $|\epsilon_i| \le q \le x_i / 2$, so that $x_i' > 0$. Finally, since $k_i' \ge k_i$,

$$x_i' \le x_i \le c_i k_i \le c_i k_i' \ .$$

Therefore, $x_i' \in (0, k_i' c_i]$ so that $y_i' = x_i'$ and

$$\begin{aligned} k_{i+1}' &= k_i' - x_i' / c_i \\ &= k_i + (k_i' - k_i) - x_i / c_i - \epsilon_i / c_i \\ &= k_{i+1} + (k_i' - k_i) - \epsilon_i / c_i \\ &= k_{i+1} + \sum_{j \le i,\ x_j \in (0, k_j c_j]} -\epsilon_j / c_j, \end{aligned}$$

where the last equality uses the induction assumption. Now, since $\epsilon_j \leq 0$ for all j, this implies $k'_{i+1} \geq k_{i+1}$. Further, $|\epsilon_i/c_i| \leq dc_N/(Nc_i) \leq d/N$ and by the inductive assumption, $|k'_i - k_i| \leq d\frac{i-1}{N}$ so that $|k'_{i+1} - k_{i+1}| \leq d\frac{i}{N}$ as desired. Finally, since $y'_i = x'_i$ and $y_i = x_i$, $|y'_i - x'_i| = 0 = |y_i - x_i|$.

Case 2 ($i = M$): First we show that $y'_i = k'_i c_i$, which implies $k'_{i+1} = 0$, and then we prove the expression for $|y'_i - x'_i|$. Since $x_M \neq k_M c_M$, we must have either either $x_i > k_i c_i$ or $M = N$.

If $M = N$, then the claim $y'_i = k'_i c_i$ is immediate by inspection of Algorithm 9. So suppose $x_i > k_i c_i$. By the inductive assumption, $k'_i \leq k_i + d\frac{i}{N} \leq k_i + d$. Now, we observe that $d \leq \frac{1}{2c_1}(x_i - c_i k_i) \leq \frac{1}{2c_i}(x_i - c_i k_i)$, which implies

$$\begin{aligned} c_i k'_i &\leq c_i k_i + c_i d \\ &\leq c_i k_i + (x_i - c_i k_i)/2 \\ &\leq x_i - (x_i - c_i k_i)/2 \,. \end{aligned}$$

Next, observe that $d \leq \frac{1}{2}(x_i - c_i k_i)$ to conclude

$$\begin{aligned} c_i k'_i &\leq x_i - (x_i - c_i k_i)/2 \\ &\leq x_i - d \\ &\leq x_i - h \\ &\leq x'_i \,. \end{aligned}$$

Therefore, $x'_i \geq k'_i c_i$, so that $y'_i = c_i k'_i$.

It remains to compute $|y'_i - x'_i|$. By the induction assumption, we have

$$k'_i = k_i + \sum_{j<i,\ x_j \in (0, k_j c_j]} -\epsilon_j/c_j \,.$$

Therefore,

$$x'_i - y'_i = x_i + \epsilon_M - y_i + c_M \sum_{j<i,\ x_j \in (0, k_j c_j]} \epsilon_j/c_j \,. \tag{6}$$

Observe that $\epsilon_M + c_M \sum_{j<i,\ x_j \in (0, k_j c_j]} \epsilon_j/c_j \leq 0$ since $\epsilon_i \leq 0$ for all $i \leq M$. Now, since $c_M \leq c_j$ for $j \leq M$, we have

$$\left| \epsilon_M + c_M \sum_{j<i,\ x_j \in (0, k_j c_j]} \epsilon_j/c_j \right| \leq Nh \leq d \,.$$

Now, since $x_M \neq x_M k_M$, and $i = M$, we have $d \leq \frac{|x_i - c_i k_i|}{2}$ by definition so that

$$\left| \epsilon_M + c_M \sum_{j<i,\ x_j \in (0, k_j c_j]} \epsilon_j/c_j \right| \leq |x_i - c_i k_i|/2 = \frac{|x_i - y_i|}{2} \,.$$

Now, recalling equation (6) we have

$$\begin{aligned}\operatorname{sign}(x_i' - y_i') &= \operatorname{sign}\left(x_i - y_i + \left[\epsilon_M + c_M \sum_{j<i,\ x_j \in (0, k_j c_j]} \epsilon_j / c_j\right]\right) \\ &= \operatorname{sign}(x_i - y_i),\end{aligned}$$

where in the last line we have used $\left|\epsilon_M + c_M \sum_{j<i,\ x_j \in (0,k_j c_j]} \epsilon_j / c_j\right| \leq \frac{|x_i - y_i|}{2}$. Therefore, we have

$$\begin{aligned}|x_i' - y_i'| &= \operatorname{sign}(x_i' - y_i')(x_i' - y_i') \\ &= \operatorname{sign}(x_i - y_i)\left(x_i - y_i + \epsilon_M + c_M \sum_{j<i,\ x_j \in (0,k_j c_j]} \epsilon_j / c_j\right) \\ &= |x_i - y_i| + \operatorname{sign}(x_i - y_i)\left(\epsilon_M + c_M \sum_{j<i,\ x_j \in (0,k_j c_j]} \epsilon_j / c_j\right).\end{aligned}$$

Case 3 ($i > M$):

Since $k_i' = 0$ by inductive hypothesis, we must have $y_i' = 0$ as desired. Further, observe that as observed in the beginning of the proof, $k_i = 0$ for all $i > M$ as well so that we have $y_i = 0$. Finally, if $x_i > 0$, we have $x_i + \epsilon_i \geq x_i/2 > 0$ since $|\epsilon_i| \leq q \leq x_i/2$ so that $\operatorname{sign}(x_i') = \operatorname{sign}(x_i)$. Therefore, we can conclude

$$|y_i' - x_i'| = |x_i'| = \begin{cases} |x_i| - \epsilon_i & x_i \leq 0 \\ |x_i| + \epsilon_i & x_i > 0 \end{cases}.$$

Since $y_i = 0$, $|x_i| = |y_i - x_i|$ and this is the desired form for $|y_i' - x_i'|$.

This concludes the induction.

From the expression for $|y_i' - x_i'|$ we see that if g is given by

$$g_i = \begin{cases} -1 & x_i \leq 0 \\ 1 & x_i > k_i c_i \\ \operatorname{sign}(x_M - y_M)\frac{c_M}{c_i} & x_i \in (0, k_i c_i],\ x_M \neq k_M c_M \\ \frac{c_M}{c_i} & x_i \in (0, k_i c_i],\ x_M = k_M c_M \end{cases}$$

then $S_W(x + \epsilon) = S_W(x) + \langle g, \epsilon \rangle$. Finally, observe that our perturbation x_δ has the property $\operatorname{sign}((x_\delta)_M - y_M) = 1$ if $x_M = k_M y_M$ to prove the Proposition. ■

Appendix F. Proof of Theorem 7

We re-state Theorem 7 below for reference:

Theorem 7 *Let $\mathcal{A}$ be an online linear optimization algorithm that outputs w_t in response to g_t. Suppose W is a convex closed set of diameter D. Suppose $\mathcal{A}$ guarantees for all t and $\mathring{v}$:*

$$\sum_{i=1}^{t}\langle \tilde{g}_i, w_i - \mathring{v}\rangle \leq \epsilon + \|\mathring{v}\|A\sqrt{\sum_{i=1}^{t}\|\tilde{g}_i\|_\star^2\left(1+\ln\left(\frac{\|\mathring{v}\|^2 t^C}{\epsilon^2}+1\right)\right)} + B\|\mathring{v}\|\ln\left(\frac{\|\mathring{v}\|t^C}{\epsilon}+1\right),$$

for constants A, B and C and ϵ independent of t. Then for all $\mathring{w}\in W$, Algorithm 6 guarantees

$$R_T(\mathring{w}) \leq \sum_{t=1}^{T}\langle g_t, x_t - \mathring{w}\rangle \leq O\left(\sqrt{V_T(\mathring{w})\ln\tfrac{TD}{\epsilon}\ln(T)} + \ln\tfrac{DT}{\epsilon}\ln(T) + \epsilon\right),$$

where $V_T(\mathring{w}) := \|\overline{x}_0 - \mathring{w}\|^2 + \sum_{t=1}^{T}\|\tilde{g}_t\|_\star^2\|x_t-\mathring{w}\|^2 \leq D^2 + \sum_{t=1}^{T}\|g_t\|_\star^2\|x_t-\mathring{w}\|^2$.

Proof For any t, consider the random vector X_t that takes value x_i for $i \leq t$ with probability proportional to $\|\tilde{g}_i\|_\star^2$ and value $\overline{x}_0$ with probability proportional to 1. Make the following definitions/observations:

1. $Z_t := 1 + \sum_{i=1}^{t}\|\tilde{g}_i\|_\star^2$ for all t, so that
$$V_T(\mathring{w}) = \|\overline{x}_0 - \mathring{w}\|^2 + \sum_{t=1}^{T}\|\tilde{g}_t\|_\star^2\|x_t-\mathring{w}\|^2 = Z_T\mathbb{E}[\|X_T-\mathring{w}\|^2]\,.$$
2. $\overline{x}_T = \mathbb{E}[X_T] = \frac{\overline{x}_0+\sum_{t=1}^{T}\|\tilde{g}_t\|_\star^2 x_t}{1+\sum_{t=1}^{T}\|\tilde{g}_t\|_\star^2}$.
3. $\sigma_t^2 := \frac{\|\overline{x}_t-\overline{x}_0\|^2+\sum_{i=1}^{t}\|\tilde{g}_i\|_\star^2\|x_i-\overline{x}_t\|^2}{Z_t}$ so that $\sigma_t^2 = \mathbb{E}[\|X_t-\overline{x}_t\|^2]$, and $\sigma_T^2 Z_T = \|\overline{x}_0-\overline{x}_T\|^2 + \sum_{t=1}^{T}\|\tilde{g}_t\|_\star^2\|x_t-\overline{x}_T\|^2$.

To prove the theorem, we are going to show for any $\mathring{w}\in W$,

$$R_T(\mathring{w}) \leq O\left[\sqrt{Z_T\|\mathring{w}-\overline{x}_T\|^2\ln\frac{TD}{\epsilon^2}} + \ln\frac{DT}{\epsilon}\ln(T) + \sqrt{Z_T\sigma_T^2\ln\frac{TD}{\epsilon}\log(T)}\right], \tag{7}$$

which implies the desired bound by a bias-variance decomposition: $Z_T\|\mathring{w}-\overline{x}_T\|^2 + Z_T\sigma_T^2 = Z_T\mathbb{E}[\|X_T-\mathring{w}\|^2] = V_T(\mathring{w})$.

Observe that, by triangle inequality and the definition of dual norm, $\langle g_t, z\rangle + \|g_t\|_\star S_W(z) \geq \langle g_t, x\rangle$ for all z and $x\in\Pi_W(z)$, with equality when $z\in W$. Hence, we have

$$\langle g_t, x_t-\mathring{w}\rangle \leq \langle g_t, z_t-\mathring{w}\rangle + \|g_t\|_\star S_W(z_t) - \|g_t\|_\star S_W(\mathring{w}) \leq \langle \tilde{g}_t, z_t-\mathring{w}\rangle, \tag{8}$$

for all $\mathring{w}\in W$, where in the last inequality we used Proposition 1. Using this inequality with the regret guarantee of $\mathcal{A}$, we have

$$\begin{aligned}
R_T(\mathring{w}) &\leq \sum_{t=1}^{T}\langle g_t, x_t-\mathring{w}\rangle \leq \sum_{t=1}^{T}\langle \tilde{g}_t, z_t-\mathring{w}\rangle = \sum_{t=1}^{T}\langle \tilde{g}_t, w_t-(\mathring{w}-\overline{x}_T)\rangle + \sum_{t=1}^{T}\langle \tilde{g}_t, \overline{x}_{t-1}-\overline{x}_T\rangle\\
&\leq O\left(\|\mathring{w}-\overline{x}_T\|\sqrt{\sum_{t=1}^{T}\|\tilde{g}_t\|_\star^2\ln\frac{\|\mathring{w}-\overline{x}_T\|T}{\epsilon^2}} + \|\mathring{w}-\overline{x}_T\|\ln\frac{\|\mathring{w}-\overline{x}_T\|T}{\epsilon}\right) + \epsilon + \sum_{t=1}^{T}\langle \tilde{g}_t, \overline{x}_{t-1}-\overline{x}_T\rangle\\
&= O\left(\sqrt{Z_T\|\mathring{w}-\overline{x}_T\|^2\ln\frac{DT}{\epsilon^2}} + D\ln\frac{DT}{\epsilon}\right) + \epsilon + \sum_{t=1}^{T}\langle \tilde{g}_t, \overline{x}_{t-1}-\overline{x}_T\rangle\,.
\end{aligned}$$

Note that the first term is exactly what we want, so we only have to upper bound the second one. This is readily done through Lemma 22 that immediately gives us the stated result. ■

Lemma 22 *Under the hypotheses of Theorem 7, we have*

$$\sum_{t=1}^{T}\langle \tilde{g}_t, \overline{x}_{t-1} - \overline{x}_T\rangle \leq M\sqrt{Z_T}\sigma_T\sqrt{1+\ln Z_T} + K(1+\ln Z_T),$$

where $M = A\sqrt{1+\ln\left(\frac{2D^2T^C}{\epsilon^2}+3T^C\right)}$ *and* $K = 1 + B\ln\left(\frac{\sum_{t=1}^{T}\|g_t\|_\star DT^C}{\epsilon}+2T^C\right)$.

Proof We have that

$$\sum_{i=1}^{t}\langle \tilde{g}_i, \overline{x}_{i-1} - \overline{x}_t\rangle - \sum_{i=1}^{t-1}\langle \tilde{g}_i, \overline{x}_{i-1} - \overline{x}_{t-1}\rangle = \left\langle \sum_{i=1}^{t}\tilde{g}_i, \overline{x}_{t-1} - \overline{x}_t\right\rangle .$$

The telescoping sum gives us

$$\sum_{t=1}^{T}\langle \tilde{g}_t, \overline{x}_{t-1} - \overline{x}_T\rangle = \sum_{t=1}^{T}\left\langle \sum_{i=1}^{t}\tilde{g}_i, \overline{x}_{t-1} - \overline{x}_t\right\rangle \leq \sum_{t=1}^{T}\left\|\sum_{i=1}^{t}\tilde{g}_i\right\|_\star \|\overline{x}_{t-1} - \overline{x}_t\| .$$

So in order to bound $\sum_{t=1}^{T}\langle \tilde{g}_t, \overline{x}_{t-1} - \overline{x}_T\rangle$, it suffices to bound $\left\|\sum_{i=1}^{t}\tilde{g}_i\right\|_\star \|\overline{x}_{t-1} - \overline{x}_t\|$ by a sufficiently small value. First we will tackle $\left\|\sum_{i=1}^{t}\tilde{g}_i\right\|$. To do this we recall our regret bound for $\mathcal{A}$. Analogous to (8), we have

$$\begin{aligned}\langle g_t, x_t\rangle &\geq \langle g_t, z_t\rangle + \|g_t\|_\star S_W(z_t) + \langle \tilde{g}_t, x_t - z_t\rangle\\ \langle \tilde{g}_t, z_t\rangle &\geq \langle g_t, z_t - x_t\rangle + \|g_t\|_\star\|z_t - x_t\| + \langle \tilde{g}_t, x_t\rangle\\ &\geq \langle \tilde{g}_t, x_t\rangle .\end{aligned}$$

Therefore, for any $X \in \mathbb{R}$ we have:

$$\begin{aligned}&\sum_{i=1}^{t} - \|\tilde{g}_i\|_\star D + \left\|\sum_{i=1}^{t}\tilde{g}_i\right\|_\star X\\ &\quad\leq \sum_{i=1}^{t}\langle \tilde{g}_i, x_i - \overline{x}_{i-1}\rangle + \left\|\sum_{i=1}^{t}\tilde{g}_i\right\|_\star X\\ &\quad\leq \sum_{i=1}^{t}\langle \tilde{g}_i, z_i - \overline{x}_{i-1}\rangle + \left\|\sum_{i=1}^{t}\tilde{g}_i\right\|_\star X\\ &\quad= \sum_{i=1}^{t}\langle \tilde{g}_i, w_i\rangle + \left\|\sum_{i=1}^{t}\tilde{g}_i\right\|_\star X\\ &\quad\leq \epsilon + |X|A\sqrt{\sum_{i=1}^{t}\|\tilde{g}_i\|_\star^2\left(1+\ln\left(\frac{|X|^2t^C}{\epsilon^2}+1\right)\right)} + B|X|\ln\left(\frac{|X|t^C}{\epsilon}+1\right),\end{aligned}$$

where in the first inequality we have used the fact that the domain is bounded.

Dividing by X and solving for $\left\|\sum_{i=1}^t \tilde{g}_i\right\|_\star$, we have

$$\left\|\sum_{i=1}^t \tilde{g}_i\right\|_\star \leq \frac{\epsilon}{X} + A\sqrt{\sum_{i=1}^t \|\tilde{g}_i\|_\star^2 \left(1 + \ln\left(\frac{|X|^2 t^C}{\epsilon^2} + 1\right)\right)} + B\ln\left(\frac{|X|t^C}{\epsilon} + 1\right) + \frac{\sum_{i=1}^t \|\tilde{g}_i\|_\star D}{X} \,.$$

Set $X = \epsilon + \sum_{i=1}^t \|\tilde{g}_i\|_\star D$ and over-approximate to conclude:

$$\begin{aligned}\left\|\sum_{i=1}^t \tilde{g}_i\right\|_\star &\leq 1 + A\sqrt{\sum_{i=1}^t \|\tilde{g}_i\|_\star^2 \left(1 + \ln\left(\frac{2D^2\left(\sum_{i=1}^t \|\tilde{g}_i\|_\star\right)^2 t^C}{\epsilon^2} + 3t^C\right)\right)} \\ &\quad + B\ln\left(\frac{\sum_{i=1}^t \|\tilde{g}_i\|_\star D t^C}{\epsilon} + 2t^C\right) \\ &\leq M\sqrt{\sum_{i=1}^t \|\tilde{g}_i\|_\star^2} + K \,.\end{aligned}$$

With this in hand, we have

$$\sum_{t=1}^T \langle \tilde{g}_t, \overline{x}_{t-1} - \overline{x}_T\rangle \leq \sum_{t=1}^T \left\|\sum_{i=1}^t \tilde{g}_i\right\|_\star \|\overline{x}_{t-1} - \overline{x}_t\| \leq M\sum_{t=1}^T \sqrt{\sum_{i=1}^t \|\tilde{g}_i\|_\star^2}\|\overline{x}_{t-1} - \overline{x}_t\| + K\sum_{t=1}^T \|\overline{x}_{t-1} - \overline{x}_t\| \,. \tag{9}$$

Now, we relate $\|\overline{x}_t - \overline{x}_{t-1}\|$ to $\|x_t - \overline{x}_t\|$:

$$\overline{x}_{t-1} - \overline{x}_t = \overline{x}_{t-1} - \frac{Z_{t-1}\overline{x}_{t-1} + \|\tilde{g}_t\|_\star^2 x_t}{Z_t} = \frac{\|\tilde{g}_t\|_\star^2}{Z_t}(\overline{x}_{t-1} - x_t) = \frac{\|\tilde{g}_t\|_\star^2}{Z_t}(\overline{x}_t - x_t) + \frac{\|\tilde{g}_t\|_\star^2}{Z_t}(\overline{x}_{t-1} - \overline{x}_t),$$

that implies

$$Z_t(\overline{x}_{t-1} - \overline{x}_t) = \|\tilde{g}_t\|_\star^2(x_t - \overline{x}_t) + \|\tilde{g}_t\|_\star^2(\overline{x}_{t-1} - \overline{x}_t),$$

that is

$$\overline{x}_{t-1} - \overline{x}_t = \frac{\|\tilde{g}_t\|_\star^2}{Z_{t-1}}(x_t - \overline{x}_t) \,. \tag{10}$$

Hence, we have

$$M\sum_{t=1}^T \sqrt{\sum_{i=1}^t \|\tilde{g}_i\|_\star^2}\|\overline{x}_t - \overline{x}_{t-1}\| \leq M\sum_{t=1}^T \sqrt{Z_t}\frac{\|g_t\|_\star^2}{Z_{t-1}}\|x_t - \overline{x}_t\|,$$

and

$$K\sum_{t=1}^T \|\overline{x}_t - \overline{x}_{t-1}\| \leq K\sum_{t=1}^T \frac{\|g_t\|_\star^2}{Z_{t-1}}\|x_t - \overline{x}_t\| \leq KD\sum_{t=1}^T \frac{\|g_t\|_\star^2}{Z_{t-1}} \,.$$

Using Cauchy–Schwarz inequality, we have

$$M\sum_{t=1}^T \sqrt{Z_t}\frac{\|g_t\|_\star^2}{Z_{t-1}}\|x_t - \overline{x}_t\| \leq M\sqrt{\sum_{t=1}^T \frac{\|\tilde{g}_t\|_\star^2}{Z_{t-1}}}\sqrt{\sum_{t=1}^T \frac{Z_t}{Z_{t-1}}\|\tilde{g}_t\|_\star^2\|x_t - \overline{x}_t\|^2} \,.$$

So, putting together the last inequalities, we have

$$\sum_{t=1}^{T}\langle \tilde{g}_t, \overline{x}_{t-1} - \overline{x}_T\rangle \leq M\sqrt{\sum_{t=1}^{T}\frac{\|\tilde{g}_t\|_\star^2}{Z_{t-1}}}\sqrt{\sum_{t=1}^{T}\frac{Z_t}{Z_{t-1}}\|\tilde{g}_t\|_\star^2\|x_t-\overline{x}_t\|^2} + KD\sum_{t=1}^{T}\frac{\|g_t\|_\star^2}{Z_{t-1}}\,.$$

We now focus on the the term $\sum_{t=1}^{T}\frac{\|g_t\|_\star^2}{Z_{t-1}}$ that is easily bounded:

$$\begin{aligned}
\sum_{t=1}^{T}\frac{\|g_t\|_\star^2}{Z_{t-1}} &= \sum_{t=1}^{T}\left(\frac{\|\tilde{g}_t\|_\star^2}{Z_t} + \frac{\|\tilde{g}_t\|_\star^2}{Z_{t-1}} - \frac{\|\tilde{g}_t\|_\star^2}{Z_t}\right)\\
&\leq \sum_{t=1}^{T}\left(\frac{\|\tilde{g}_t\|_\star^2}{Z_t} + \frac{1}{Z_{t-1}} - \frac{1}{Z_t}\right)\\
&\leq \frac{1}{Z_0} + \sum_{t=1}^{T}\frac{\|\tilde{g}_t\|_\star^2}{Z_t}\\
&\leq \frac{1}{Z_0} + \log\frac{Z_T}{Z_0}\\
&= 1 + \ln Z_T,
\end{aligned}$$

where in the last inequality we used the well-known inequality $\sum_{t=1}^{T}\frac{a_t}{a_0+\sum_{i=1}^{t}a_i} \leq \ln(1+\frac{\sum_{t=1}^{T}a_t}{a_0})$, $\forall a_t \geq 0$.

To upper bound the term $\sum_{t=1}^{T}\frac{Z_t}{Z_{t-1}}\|\tilde{g}_t\|_\star^2\|x_t-\overline{x}_t\|^2$, observe that

$$\begin{aligned}
\sigma_T^2 Z_T &= \|\overline{x}_0 - \overline{x}_T\|^2 + \sum_{t=1}^{T}\|\tilde{g}_t\|_\star^2\|x_t-\overline{x}_T\|^2\\
&= \|\overline{x}_0 - \overline{x}_T\|^2 + \sum_{t=1}^{T-1}\|\tilde{g}_t\|_\star^2\|x_t-\overline{x}_T\|^2 + \|\tilde{g}_T\|_\star^2\|x_T-\overline{x}_T\|^2\\
&= Z_{T-1}(\sigma_{T-1}^2 + \|\overline{x}_T - \overline{x}_{T-1}\|^2) + \|\tilde{g}_T\|_\star^2\|x_T-\overline{x}_T\|^2\\
&= Z_{T-1}\sigma_{T-1}^2 + \|\tilde{g}_T\|_\star^2\left(1+\frac{\|\tilde{g}_T\|_\star^2}{Z_{T-1}}\right)\|x_T-\overline{x}_T\|^2\\
&= Z_{T-1}\sigma_{T-1}^2 + \|\tilde{g}_T\|_\star^2\frac{Z_T}{Z_{T-1}}\|x_T-\overline{x}_T\|^2,
\end{aligned}$$

where the third equality comes from bias-variance decomposition and the fourth one comes from (10). Hence, we have

$$\sum_{t=1}^{T}\frac{Z_t}{Z_{t-1}}\|\tilde{g}_t\|_\star^2\|x_t-\overline{x}_t\|^2 = \sum_{t=1}^{T}(\sigma_t^2 Z_t - \sigma_{t-1}^2 Z_{t-1}) \leq \sigma_T^2 Z_T\,.$$

Putting all together, we have the stated bound. ■