Nonparametric Online Regression while Learning the Metric

Ilja Kuzborskij
EPFL
Switzerland
ilja.kuzborskij@gmail.com

Nicolò Cesa-Bianchi
Dipartimento di Informatica
Università degli Studi di Milano
Milano 20135, Italy
nicolo.cesa-bianchi@unimi.it

October 24, 2017

Abstract

We study algorithms for online nonparametric regression that learn the directions along which
the regression function is smoother. Our algorithm learns the Mahalanobis metric based on the
gradient outer product matrix \( G \) of the regression function (automatically adapting to the
effective rank of this matrix), while simultaneously bounding the regret —on the same data sequence—
in terms of the spectrum of \( G \). As a preliminary step in our analysis, we extend a nonparametric online
learning algorithm by Hazan and Megiddo enabling it to compete against functions whose
Lipschitzness is measured with respect to an arbitrary Mahalanobis metric.

1 Introduction

An online learner is an agent interacting with an unknown and arbitrary environment over a sequence
of rounds. At each round \( t \), the learner observes a data point (or instance) \( x_t \in \mathcal{X} \subseteq \mathbb{R}^d \),
outputs a prediction \( \hat{y}_t \) for the label \( y_t \in \mathbb{R} \) associated with that instance, and incurs some loss \( \ell_t(\hat{y}_t) \),
which in this paper is the square loss \((\hat{y}_t - y_t)^2\). At the end of the round, the label \( y_t \) is given to the
learner, which he can use to reduce his loss in subsequent rounds. The performance of an online learner
is typically measured using the regret. This is defined as the amount by which the learner’s cumulative
loss exceeds the cumulative loss (on the same sequence of instances and labels) of any function \( f \)
in a given reference class \( \mathcal{F} \) of functions,

\[
R_T(f) = \sum_{t=1}^{T} \left( \ell_t(\hat{y}_t) - \ell_t(f(x_t)) \right) \quad \forall f \in \mathcal{F}.
\]

(1)

Note that typical regret bounds apply to all \( f \in \mathcal{F} \) and to all individual data sequences. However, the
bounds are allowed to scale with parameters arising from the interplay between \( f \) and the data sequence.

In order to capture complex environments, the reference class of functions should be large. In this
work we focus on nonparametric classes \( \mathcal{F} \), containing all differentiable functions that are smooth with
respect to some metric on \( \mathcal{X} \). Our approach builds on the simple and versatile algorithm for nonparametric
online learning introduced in [7]. This algorithm has a bound on the regret \( R_T(f) \) of order (ignoring
logarithmic factors)

\[
\left( 1 + \sqrt{\sum_{i=1}^{d} \|\partial_i f\|_\infty^2} \right)^{T + \frac{d}{2}} \quad \forall f \in \mathcal{F}.
\]

(2)

Here \( \|\partial_i f\|_\infty \) is the value of the partial derivative \( \partial f(x) / \partial x_i \) maximized over \( x \in \mathcal{X} \). The square
root term is the Lipschitz constant of \( f \), measuring smoothness with respect to the Euclidean metric.
However, in some directions \( f \) may be smoother than in others. Therefore, if we knew in advance the
set of directions along which the best performing reference functions \( f \) are smooth, we could use this
information to control regret better. In this paper we extend the algorithm from [7] and make it adaptive
to the Mahalanobis distance defined through an arbitrary positive definite matrix $M$ with spectrum \(\{(u_i, \lambda_i)\}_{i=1}^d\) and unit spectral radius ($\lambda_1 = 1$). We prove a bound on the regret $R_T(f)$ of order (ignoring logarithmic factors)

\[
\sqrt{\det(M)} + \sum_{i=1}^d \frac{\|\nabla u_i f\|_\infty^2}{\lambda_i} T^{1 + \rho_T} \quad \forall f \in \mathcal{F}.
\]

Here $\rho_T \leq d$ is, roughly, the number of eigenvalues of $M$ larger than a threshold shrinking polynomially in $T$, and $\det(M) \leq 1$ is the determinant of $M$ truncated at $\lambda_\kappa$ (with $\kappa \leq \rho_T$). The quantity $\|\nabla u_i f\|_\infty$ is defined like $\|\partial_i f\|_\infty$, but with the directional derivative $\nabla f(x)^T u_i$ instead of the partial derivative. When the spectrum of $M$ is light-tailed (so that $\rho_T \ll d$ and, simultaneously, $\det(M) \ll 1$), with the smaller eigenvalues $\lambda_i$ corresponding to eigenvectors in which $f$ is smoother (so that the ratios $\|\nabla u_i f\|_\infty / \lambda_i$ remain controlled), then our bound improves on (2). On the other hand, when no preliminary knowledge about good $f$ is available, we may run the algorithm with $M$ equal to the identity matrix and recover exactly the bound (2).

Given that the regret can be improved by informed choices of $M$, it is natural to ask whether some kind of improvement is still possible when $M$ is learned online, from the same data sequence on which the regret is being measured. Of course, this question makes sense if the data tell us something about the smoothness of the $f$ against which we are measuring the regret. In the second part of the paper we implement this idea by considering a scenario where instances are drawn i.i.d. from some unknown distribution, labels are stochastically generated by some unknown regression function $f_0$, and we have no preliminary knowledge about the directions along which $f_0$ is smoother.

In this stochastic scenario, the expected gradient outer product matrix $G = \mathbb{E} \left[ (\nabla f_0(X) \nabla f_0(X)^\top) \right]$ provides a natural choice for the matrix $M$ in our algorithm. Indeed, $\mathbb{E} \left[ (\nabla f_0(X)^\top u_i)^2 \right] = \mu_i$, where $u_1, \ldots, u_d$ are the eigenvectors of $G$ while $\mu_1, \ldots, \mu_d$ are the corresponding eigenvalues. Thus, eigenvectors $u_1, \ldots, u_d$ capture the principal directions of variation for $f$. In fact, assuming that the labels obey a statistical model $Y = g(BX) + \varepsilon$ where $\varepsilon$ is the noise and $B \in \mathbb{R}^{k \times d}$ projects $X$ onto a $k$-dimensional subspace of $\mathcal{X}$, one can show [23] that $\text{span}(B) \equiv \text{span}(u_1, \ldots, u_d)$. In this sense, $G$ is the “best” metric, because it recovers the $k$-dimensional relevant subspace.

When $G$ is unknown, we run our algorithm in phases using a recently proposed estimator $\hat{G}$ of $G$. The estimator is trained on the same data sequence and is fed to the algorithm at the beginning of each phase. Under mild assumptions on $f_0$, the noise in the labels, and the instance distribution, we prove a high probability bound on the regret $R_T(f_0)$ of order (ignoring logarithmic factors)

\[
1 + \sum_{j=1}^d \left( \frac{\|\nabla u_j f_0\|_\infty}{\mu_j} + \frac{\|\nabla V f_0\|_\infty}{\mu_1} \right)^2 T^{1 + \tilde{\rho}_T}.
\]

Observe that the rate at which the regret grows is the same as the one in (3), though now the effective dimension parameter $\tilde{\rho}_T$ is larger than $\rho_T$ by an amount related to the rate of convergence of the eigenvalues of $\hat{G}$ to those of $G$. The square root term is also similar to (3), but for the extra quantity $\|\nabla V f_0\|_\infty$, which accounts for the error in approximating the eigenvectors of $G$. More precisely, $\|\nabla V f_0\|_\infty$ is $\|\nabla u f\|_\infty$ maximized over directions $v$ in the span of $V$, where $V$ contains those eigenvectors of $G$ that cannot be identified because their eigenvalues are too close to each other (we come back to this issue shortly). Finally, we lose the dependence on the truncated determinant, which is replaced here by its trivial upper bound 1.

The proof of (2) in [7] is based on the sequential construction of a sphere packing of $\mathcal{X}$, where the spheres are centered on adaptively chosen instances $x_t$, and have radii shrinking polynomially with time. Each sphere hosts an online learner, and each new incoming instance is predicted using the learner hosted in the nearest sphere. Our variant of that algorithm uses an ellipsoid packing, and computes distances using the Mahalanobis distance $\|\cdot\|_M$. The main new ingredient in the analysis leading to (3) is our notion of effective dimension $\rho_T$ (we call it the effective rank of $M$), which measures how fast the spectrum
who considered regression problems in large spaces and proved bounds on the regret. Minimax rates
This loss is high whenever dissimilar pairs of
variant that automatically adapts to the intrinsic dimension of the data manifold. Our work explores an
2 Related works
Given a positive definite
(2)
sequence
Preliminaries and notation.
We consider the following online learning protocol with oblivious adversary. Given an unknown
sequence
(3)
vector
(4)
M
vanishes. The proof also uses an ellipsoid packing bound and a lemma relating the Lipschitz constant
to the Mahalanobis distance.

The proof of (4) is more intricate because
is only known up to a certain approximation. We use an
estimator
, recently proposed in [15], which is consistent under mild distributional assumptions when
is continuously differentiable. The first source of difficulty is adjusting the notion of effective rank (which
the algorithm needs to compute) to compensate for the uncertainty in the knowledge of the eigenvalues
of
. A further problematic issue arises because we want to measure the smoothness of
along the
eigendirections of
, and so we need to control the convergence of the eigenvectors, given that
converges to
in spectral norm. However, when two eigenvalues of
are close, then the corresponding
eigenvectors in the estimated matrix
are strongly affected by the stochastic perturbation (a phenomenon
known as hybridization or spectral leaking in matrix perturbation theory, see [1, Section 2]). Hence, in
our analysis we need to separate out the eigenvectors that correspond to well spaced eigenvalues from the
others. This lack of discrimination causes the appearance in the regret of the extra term
.

2 Related works
Nonparametric estimation problems have been a long-standing topic of study in statistics, where one
is concerned with the recovery of an optimal function from a rich class under appropriate probabilistic
assumptions. In online learning, the nonparametric approach was investigated in [17, 16, 18] by Yovk,
who considered regression problems in large spaces and proved bounds on the regret. Minimax rates
for the regret were later derived in [14] using a non-constructive approach. The first explicit online
nonparametric algorithms for regression with minimax rates were obtained in [5].

The nonparametric online algorithm of [7] is known to have a suboptimal regret bound for Lipschitz
classes of functions. However, it is a simple and efficient algorithm, well suited to the design of extensions
that exhibit different forms of adaptivity to the data sequence. For example, the paper [10] derived a
variant that automatically adapts to the intrinsic dimension of the data manifold. Our work explores an
alternative direction of adaptivity, mainly aimed at taming the effect of the curse of dimensionality in
nonparametric prediction through the learning of an appropriate Mahalanobis distance on the instance
space. There is a rich literature on metric learning (see, e.g., the survey [2]) where the Mahalanobis metric
is typically learned through minimization of the pairwise loss function of the form
. This loss is high whenever dissimilar pairs of
and
are close in the Mahalanobis metric, and whenever similar ones are far apart in the same metric —see, e.g., [21]. The works [6, 8, 20] analyzed generalization and consistency properties of online learning algorithms employing pairwise losses.

In this work we are primarily interested in using a metric
where
is close to the gradient
outer product matrix of the best model in the reference class of functions. As we are not aware whether
pairwise loss functions can indeed consistently recover such metrics, we directly estimate the gradient
outer product matrix. This approach to metric learning was mostly explored in statistics —e.g., by locally-
linear Empirical Risk Minimization on RKHS [13, 12], and through Stochastic Gradient Descent [4]. Our
learning approach combines —in a phased manner— a Mahalanobis metric extension of the algorithm
by [7] with the estimator of [15]. Our work is also similar in spirit to the “gradient weights” approach
of [9], which learns a distance based on a simpler diagonal matrix.

Preliminaries and notation. Let
be the ball of center
and radius
. We assume instances
belong to
and labels
belong to
.

We consider the following online learning protocol with oblivious adversary. Given an unknown
sequence
, the environment reveals instance
; the learner selects an action
and incurs the square loss
; the learner observes
.

Given a positive definite
matrix
, the norm
induced by
(a.k.a. Mahalanobis distance) is defined by
.
Definition 1 (Covering and Packing Numbers). An \( \varepsilon \)-cover of a set \( S \) w.r.t. some metric \( \rho \) is a set \( \{x_1', \ldots, x_m'\} \subseteq S \) such that for each \( x \in S \) there exists \( i \in \{1, \ldots, n\} \) such that \( \rho(x, x'_i) \leq \varepsilon \). The covering number \( \mathcal{N}(S, \varepsilon, \rho) \) is the smallest cardinality of a \( \varepsilon \)-cover.

An \( \varepsilon \)-packing of a set \( S \) w.r.t. some metric \( \rho \) is a set \( \{x_1', \ldots, x_m'\} \subseteq S \) such that for any distinct \( i, j \in \{1, \ldots, m\} \), we have \( \rho(x'_i, x'_j) > \varepsilon \). The packing number \( \mathcal{M}(S, \varepsilon, \rho) \) is the largest cardinality of a \( \varepsilon \)-packing.

It is well known that \( \mathcal{M}(S, 2\varepsilon, \rho) \leq \mathcal{N}(S, \varepsilon, \rho) \leq \mathcal{M}(S, \varepsilon, \rho) \). For all differentiable \( f : \mathcal{X} \to \mathcal{Y} \) and for any orthonormal basis \( V \equiv \{u_1, \ldots, u_k\} \) with \( k \leq d \) we define

\[
\|\nabla_V f\|_\infty = \max_{v \in \text{span}(V)} \sup_{x \in \mathcal{X}} \|\nabla f(x)^\top v\|.
\]

If \( V = \{u\} \) we simply write \( \|\nabla_u f\|_\infty \).

In the following, \( M \) is a positive definite \( d \times d \) matrix with eigenvalues \( \lambda_1 \geq \cdots \geq \lambda_d > 0 \) and eigenvectors \( u_1, \ldots, u_d \). For each \( k = 1, \ldots, d \) the truncated determinant is \( \det_k(M) = \lambda_1 \times \cdots \times \lambda_k \). The kappa function for the matrix \( M \) is defined by

\[
\kappa(r, t) = \max \left\{ m : \lambda_m \geq t^{-\frac{k}{2(r-1)}}, \ m = 1, \ldots, d \right\}
\]

for \( t \geq 1 \) and \( r = 1, \ldots, d \).

Note that \( \kappa(r+1, t) \leq \kappa(r, t) \). Now define the effective rank of \( M \) at horizon \( t \) by

\[
\rho_t = \min \left\{ r : \kappa(r, t) \leq r, \ r = 1, \ldots, d \right\}.
\]

Since \( \kappa(d, t) \leq d \) for all \( t \geq 1 \), this is a well defined quantity. Note that \( \rho_1 \leq \rho_2 \leq \cdots \leq d \). Also, \( \rho_t = d \) for all \( t \geq 1 \) when \( M \) is the \( d \times d \) identity matrix. Note that the effective rank \( \rho_t \) measures the number of eigenvalues that are larger than a threshold that shrinks with \( t \). Hence matrices \( M \) with extremely light-tailed spectra cause \( \rho_t \) to remain small even when \( t \) grows large. This behaviour is shown in Figure 1.

Throughout the paper, we use \( f = O(g) \) and \( f = \tilde{O}(g) \) to denote, respectively, \( f = O(g) \) and \( f = \tilde{O}(g) \).

### 3 Online nonparametric learning with ellipsoid packing

In this section we present a variant (Algorithm 1) of the online nonparametric regression algorithm introduced in [7]. Since our analysis is invariant to rescalings of the matrix \( M \), without loss of generality we assume \( M \) has unit spectral radius (i.e., \( \lambda_1 = 1 \)). Algorithm 1 sequentially constructs a packing of \( \mathcal{X} \) using \( M \)-ellipsoids centered on a subset of the past observed instances. At each step \( t \), the label of the current instance \( x_t \) is predicted using the average \( \hat{y}_t \) of the labels of past instances that fell inside the ellipsoid whose center \( x_s \) is closest to \( x_t \) in the Mahalanobis metric. At the end of the step, if \( x_t \) was outside of the closest ellipsoid, then a new ellipsoid is created with center \( x_t \). The radii \( \varepsilon_t \) of all ellipsoids are shrunk at rate \( t^{-1/(1+\rho_t)} \). Note that efficient (i.e., logarithmic in the number of centers) implementations of approximate nearest-neighbor search for the active center \( x_s \) exist [11].

The core idea of the proof (deferred to the supplementary material) is to maintain a trade-off between the regret contribution of the ellipsoids and an additional regret term due to the approximation of \( f \) by the Voronoi partitioning. The regret contribution of each ellipsoid is logarithmic in the number of predictions made. Since each instance is predicted by a single ellipsoid, if we ignore log factors the overall regret contribution is equal to the number of ellipsoids, which is essentially controlled by the packing number
Algorithm 1 Nonparametric online regression

**Input:** Positive definite $d \times d$ matrix $M$.

1: $S \leftarrow \emptyset$ \Comment{Centers}
2: for $t = 1, 2, \ldots$ do
3: \hspace{1em} $\varepsilon_t \leftarrow t^{-\frac{1}{1+\rho_T}}$ \Comment{Update radius}
4: \hspace{1em} Observe $x_t$
5: \hspace{1em} if $S \equiv \emptyset$ then
6: \hspace{2em} $S \leftarrow \{t\}$, $T_t \leftarrow \emptyset$ \Comment{Create initial ball}
7: \hspace{1em} end if
8: \hspace{1em} $s \leftarrow \arg \min_{s \in S} \|x_t - x_s\|_M$ \Comment{Find active center}
9: \hspace{1em} if $T_s \equiv \emptyset$ then
10: \hspace{2em} $y_t = \frac{1}{2}$
11: \hspace{1em} else
12: \hspace{2em} $\hat{y}_t \leftarrow \frac{1}{|T_s|} \sum_{t' \in T_s} y_{t'}$ \Comment{Predict using active center}
13: \hspace{1em} end if
14: \hspace{1em} Observe $y_t$
15: \hspace{1em} if $\|x_t - x_s\|_M \leq \varepsilon_t$ then
16: \hspace{2em} $T_s \leftarrow T_s \cup \{t\}$ \Comment{Update list for active center}
17: \hspace{1em} else
18: \hspace{2em} $S \leftarrow S \cup \{s\}$, $T_s \leftarrow \emptyset$ \Comment{Create new center}
19: \hspace{1em} end if
20: end for

w.r.t. the metric defined by $M$. The second regret term is due to the fact that—at any point of time—the prediction of the algorithm is constant within the Voronoi cells of $X$ induced by the current centers (recall that we predict with nearest neighbor). Hence, we pay an extra term equal to the radius of the ellipsoids times the Lipschitz constant which depends on the directional Lipschitzness of $f$ with respect to the eigenbasis of $M$.

**Theorem 1** (Regret with Fixed Metric). Suppose Algorithm 1 is run with a positive definite matrix $M$ with eigenbasis $u_1, \ldots, u_d$ and eigenvalues $1 = \lambda_1 \geq \cdots \geq \lambda_d > 0$. Then, for any differentiable $f : X \to Y$ we have that

$$R_T(f) \overset{\Delta}{=} \sqrt{\det \kappa(M)} + \sqrt{\sum_{i=1}^{d} \frac{\|\nabla u_i f\|_{\infty}}{\lambda_i}} T^{\frac{\rho_T}{1+\rho_T}}$$

where $\kappa = \kappa(\rho_T, T) \leq \rho_T \leq d$.

We first prove two technical lemmas about packings of ellipsoids.

**Lemma 1** (Volumetric packing bound). Consider a pair of norms $\|\cdot\|, \|\cdot\|'$ and let $B, B' \subset \mathbb{R}^d$ be the corresponding unit balls. Then

$$\mathcal{M}(B, \varepsilon, \|\cdot\|') \leq \frac{\text{vol}(B + \frac{\varepsilon}{2} B')}{\text{vol}(\frac{\varepsilon}{2} B')}.$$ 

**Lemma 2** (Ellipsoid packing bound). If $B$ is the unit Euclidean ball then

$$\mathcal{M}(B, \varepsilon, \|\cdot\|_M) \leq \left(\frac{8\sqrt{2}}{\varepsilon}\right)^s \prod_{i=1}^{s} \sqrt{\lambda_i} \quad \text{where} \quad s = \max\left\{ i : \sqrt{\lambda_i} \geq \varepsilon, i = 1, \ldots, d \right\}.$$
The following lemma states that whenever \( f \) has bounded partial derivatives with respect to the eigenbase of \( M \), then \( f \) is Lipschitz with respect to \( \| \cdot \|_M \).

**Lemma 3 (Bounded derivatives imply Lipschitzness in \( M \)-metric).** Let \( f : \mathcal{X} \to \mathbb{R} \) be everywhere differentiable. Then for any \( x, x' \in \mathcal{X} \),

\[
\left| f(x) - f(x') \right| \leq \| x - x' \|_M \sqrt{\sum_{i=1}^{d} \left\| \nabla u_i f \right\|_\infty^2}.
\]

### 4 Learning while learning the metric

In this section, we assume instances \( x_t \) are realizations of i.i.d. random variables \( X_t \) drawn according to some fixed and unknown distribution \( \mu \) which has a continuous density on its support \( \mathcal{X} \). We also assume labels \( y_t \) are generated according to the noise model \( y_t = f_0(x_t) + \nu(x_t) \), where \( f_0 \) is some unknown regression function and \( \nu(x) \) is a subgaussian zero-mean random variable for all \( x \in \mathcal{X} \). We then simply write \( R_T \) to denote the regret \( R_T(f_0) \). Note that \( R_T \) is now a random variable which we bound with high probability.

We now show how the nonparametric online learning algorithm (Algorithm 1) of Section 3 can be combined with an algorithm that learns an estimate

\[
\hat{G}_n = \frac{1}{n} \sum_{i=1}^{n} \hat{\nabla} f_0(x_t) \hat{\nabla} f_0(x_t)^\top
\]

of the expected outer product gradient matrix \( G = \mathbb{E} \left[ \nabla f_0(X) \nabla f_0(X)^\top \right] \). The algorithm (described in the supplementary material) is consistent under the following assumptions. Let \( \mathcal{X}(\tau) \) be \( \mathcal{X} \) blown up by a factor of \( 1 + \tau \).

**Assumption 1.**

1. There exists \( \tau_0 > 0 \) such that \( f_0 \) is continuously differentiable on \( \mathcal{X}(\tau_0) \).
2. There exists \( G > 0 \) such that \( \max_{x \in \mathcal{X}(\tau_0)} \| \nabla f_0(x) \| \leq G \).
3. The distribution \( \mu \) is full-dimensional: there exists \( C_\mu > 0 \) such that for all \( x \in \mathcal{X} \) and \( \epsilon > 0 \),
   \[
   \mu(B(x, \epsilon)) \geq C_\mu \epsilon^d.
   \]

In particular, the next lemma states that, under Assumption 1, \( \hat{G}_n \) is a consistent estimate of \( G \).

**Lemma 4 ([15, Theorem 1]).** If Assumption 1 holds, then there exists a nonnegative and nonincreasing sequence \( \{ \gamma_n \}_{n \geq 1} \) such that for all \( n \), the estimated gradient outerproduct (7) computed with parameters \( \epsilon_n > 0 \), and \( 0 < \tau_n < \tau_0 \) satisfies \( \| \hat{G}_n - G \|_2 \leq \gamma_n \) with high probability with respect do the random draw of \( X_1, \ldots, X_n \). Moreover, if \( \tau_n = \Theta(\epsilon_n^{1/4}) \), \( \epsilon_n = \Omega\left( \left( \ln n \right) \frac{2}{d} \right) \), and \( \epsilon_n = O\left( n^{-\frac{1}{2d+1}} \right) \) then \( \gamma_n \to 0 \) as \( n \to \infty \).

Our algorithm works in phases \( i = 1, 2, \ldots \) where phase \( i \) has length \( n(i) = 2^i \). Let \( T(i) = 2^{i+1} - 2 \) be the index of the last step in phase \( i \). The algorithm uses a nonincreasing regularization sequence \( \gamma_0 \geq \gamma_1 \geq \cdots > 0 \). Let \( \tilde{M}(0) = \gamma_0 I \). During each phase \( i \), the algorithm predicts the data points by running Algorithm 1 with \( M = \tilde{M}(i-1)/\| \tilde{M}(i-1) \|_2 \) (where \( \| \cdot \|_2 \) denotes the spectral norm). Simultaneously, the gradient outer product estimate (7) is trained over the same data points. At the end of phase \( i \), the current gradient outer product estimate \( \hat{G}(i) = \hat{G}_{T(i)} \) is used to form a new matrix \( \tilde{M}(i) = \hat{G}(i) + \gamma_{T(i)} I \). Algorithm 1 is then restarted in phase \( i + 1 \) with \( M = \tilde{M}(i)/\| \tilde{M}(i) \|_2 \). Note that the metric learning algorithm can be also implemented efficiently through nearest-neighbor search as explained in [15].
Let $\mu_1 \geq \mu_2 \geq \cdots \geq \mu_d$ be the eigenvalues and $u_1, \ldots, u_d$ be the eigenvectors of $G$. We define the $j$-th eigenvalue separation $\Delta_j$ by

$$\Delta_j = \min_{k \neq j} |\mu_j - \mu_k|.$$ 

For any $\Delta > 0$ define also $V_\Delta \equiv \{ u_j : |\mu_j - \mu_k| \geq \Delta, k \neq j \}$ and $V_\Delta^\perp = \{ u_1, \ldots, u_d \} \setminus V_\Delta$. Our results are expressed in terms of the effective rank (6) of $G$ at horizon $T$. However, in order to account for the error in estimating the eigenvalues of $G$, we define the effective rank $\hat{\rho}_t$ with respect to the following slight variant of the function kappa,

$$\tilde{\kappa}(r, t) = \max \left\{ m : \mu_m + 2\gamma_t \geq \mu_t t^{-1/r}, \; m = 1, \ldots, d \right\}. \; t \geq 1 \text{ and } r = 1, \ldots, d.$$

Let $\tilde{M}(i)$ be the estimated gradient outer product constructed at the end of phase $i$, and let $\hat{\mu}_1(i) + \gamma(i) \geq \cdots \geq \hat{\mu}_d(i) + \gamma(i)$ and $\hat{u}_1(i), \ldots, \hat{u}_d(i)$ be the eigenvalues and eigenvectors of $\tilde{M}(i)$, where we also write $\gamma(i)$ to denote $\gamma_T(i)$. We use $\tilde{\kappa}$ to denote the kappa function with estimated eigenvalues and $\hat{\rho}$ to denote the effective rank defined through $\tilde{\kappa}$. We start with a technical lemma.

**Lemma 5.** Let $\mu_d, \alpha > 0$ and $d \geq 1$. Then the derivative of $F(t) = (\mu_d + 2(T_0 + t)^{-\alpha}) t^{1/\alpha}$ is positive for all $t \geq 1$ when $T_0 \geq \left( \frac{d+1}{2\mu_d} \right)^{1/\alpha}$.

**Proof.** We have that $F'(t) \geq 0$ if and only if $t \leq \frac{2(T_0 + t)}{\alpha(d+1)} \left( 1 + (T_0 + t)^{\alpha} \mu_d \right)$. This is implied by

$$t \leq \frac{2\mu_d(T_0 + t)^{1+\alpha}}{\alpha(d+1)} \quad \text{or, equivalently,} \quad T_0 \geq A^{1/(1+\alpha)}t^{1/(1+\alpha)} - t$$

where $A = \alpha(d+1)/(2\mu_d)$. The right-hand side $A^{1/(1+\alpha)}t^{1/(1+\alpha)} - t$ is a concave function of $t$. Hence the maximum is found at the value of $t$ where the derivative is zero, this value satisfies

$$A^{1/(1+\alpha)} \left( 1 + \frac{1}{\alpha} \right) t^{-\alpha/(1+\alpha)} = 1 \quad \text{which solved for } t \quad \text{gives} \quad t = A^{1/\alpha} \left( 1 + \alpha \right)^{-1/(1+\alpha)/\alpha}.$$

Substituting this value of $t$ in $A^{1/(1+\alpha)}t^{1/(1+\alpha)} - t$ gives the condition $T_0 \geq A^{1/\alpha} \alpha(1 + \alpha)^{-1/(1+\alpha)/\alpha}$ which is satisfied when $T_0 \geq \left( \frac{d+1}{2\mu_d} \right)^{1/\alpha}$.

**Theorem 2.** Suppose Assumption 1 holds. If the algorithm is run with a regularization sequence $\gamma_0 = 1$ and $\gamma_t = t^{-\alpha}$ for some $\alpha > 0$ such that $\gamma_T \geq \gamma_t$ for all $t \geq (d + 1/2\mu_d)^{1/\alpha}$ and for $\gamma_1 \geq \gamma_2 \geq \cdots > 0$ satisfying Lemma 4, then for any given $\Delta > 0$

$$R_T \overset{\mathcal{O}}{=} \left( 1 + \sum_{j=1}^{d} \left( \frac{\| \nabla u_j f_0 \|_\infty + \| \nabla V_\Delta^\perp f_0 \|_\infty}{\mu_j/\mu_1} \right)^2 \frac{\hat{\rho}_T}{T_{1+\alpha/\hat{\rho}_T}} \right)^{-\hat{\rho}_T}$$

with high probability with respect to the random draw of $X_1, \ldots, X_T$.

Note that the asymptotic notation is hiding terms that depend on $1/\Delta$, hence we can not zero out the term $\| \nabla V_\Delta^\perp f_0 \|_\infty$ in the bound by taking $\Delta$ arbitrarily small.

**Proof.** Pick the smallest $i_0$ such that

$$T(i_0) \geq \left( \frac{d+1}{2\mu_d} \right)^{1/\alpha} \tag{8}$$
(we need this condition in the proof). The total regret in phases $1, 2, \ldots, i_0$ is bounded by $(d + 1/2\mu_d)^{1/\alpha} = \mathcal{O}(1)$. Let the value $\hat{\rho}_T(i)$ at the end of phase $i$ be denoted by $\hat{\rho}(i)$. By Theorem 1, the regret $R_T(i + 1)$ of Algorithm 1 in each phase $i + 1 > i_0$ is deterministically upper bounded by

$$R_T(i + 1) \leq \left(8 \ln (e 2^{i+1}) (8 \sqrt{2}) \Bar{\sigma}^{(i+1)} \right) + 4 \left(\sum_{j=1}^{d} \frac{\|\nabla \bar{a}_{j}(i) f_0\|^2}{\lambda_j(i)} \right)^{\frac{1}{2}} 2^{(i+1) \frac{p}{1+\mu_T(i+1)}} \tag{9}$$

where $\lambda_j(i) = \hat{\mu}_j(i) + \gamma(i)$. Here we used the trivial upper bound $\text{det}_n(\bar{M}_i(\gamma)/\|\bar{M}_i\|_2) \leq 1$ for all $\kappa = 1, \ldots, d$. Now assume that $\hat{\mu}_1(i) + \gamma(i) \leq (\bar{\mu}_m(i) + \gamma(i)) t^{\frac{2}{1+\rho}}$ for some $m, r \in \{1, \ldots, d\}$ and for some $t$ in phase $i + 1$. Hence, using Lemma 4 and $\gamma_t \leq \gamma_l$, we have that

$$\max_{j=1,\ldots,d} |\hat{\mu}_j(i) - \mu_j| \leq \|\hat{\mathcal{G}}(i) - \mathcal{G}\|_2 \leq \gamma(i) \quad \text{with high probability.} \tag{10}$$

where the first inequality is straightforward. Hence we may write

$$\mu_1 \leq \mu_1 - \gamma(i) + \gamma(i) \leq \hat{\mu}_1(i) + \gamma(i) \leq (\hat{\mu}_m(i) + \gamma(i)) t^{\frac{2}{1+\rho}} \leq (\mu_m + \gamma(i)) t^{\frac{2}{1+\rho}} \leq (\mu_m + 2\gamma(i)) t^{\frac{2}{1+\rho}} \tag{11}$$

Recall $\gamma(i) = T(i)^{-\alpha}$. Using Lemma 6, we observe that the derivative of

$$F(t) = \left(\mu_m + 2(T(i) + t)^{-\alpha}\right) t^{\frac{2}{1+\rho}}$$

is positive for all $t \geq 1$ when

$$T(i) \geq \left(\frac{r + 1}{2\mu_d}\right)^{1/\alpha} \geq \left(\frac{r + 1}{2\mu_m}\right)^{1/\alpha}$$

which is guaranteed by our choice (8). Hence, $(\mu_m + 2\gamma(i)) t^{\frac{2}{1+\rho}} \leq (\mu_m + 2\gamma(i, r)) T^{\frac{2}{1+\rho}}$ and so

$$\frac{\hat{\mu}_1(i) + \gamma(i)}{\mu_m(i) + \gamma(i)} \leq t^{\frac{2}{1+\rho}} \quad \text{implies} \quad \frac{\mu_1}{\mu_m + 2\gamma(i)} \leq T^{\frac{2}{1+\rho}}. \tag{12}$$

Recalling the definitions of $\tilde{\kappa}$ and $\tilde{\kappa}$, this in turn implies $\tilde{\kappa}(r, t) \leq \tilde{\kappa}(r, T)$, which also gives $\hat{\rho}_t \leq \hat{\rho}_T$ for all $t \leq T$. Next, we bound the approximation error in each individual eigenvalue of $\mathcal{G}$. By (10) we obtain, for any phase $i$ and for any $j = 1, \ldots, d$,

$$\mu_j + 2\gamma(i) \geq \mu_j + \gamma(i) + \gamma(i) \geq \hat{\mu}_j(i) + \gamma(i) \geq \mu_j - \gamma(i) + \gamma(i) \geq \mu_j.$$

Hence, bound (9) implies

$$R_T(i + 1) \leq \left(8 \ln (e 2^{i+1}) 12\tilde{\sigma}^r + 4 \left(\mu_1 + 2\gamma(i)\right) \left(\sum_{j=1}^{d} \frac{\|\nabla \bar{a}_{j}(i) f_0\|^2}{\mu_j}\right)\right)^{\frac{1}{2}} 2^{(i+1) \frac{r}{1+\rho}} \tag{11}$$

The error in approximating the eigenvectors of $\mathcal{G}$ is controlled via the following first-order eigenvector approximation result from matrix perturbation theory [22, equation (10.2)], for any vector $v$ of constant norm,

$$v^\top (\hat{a}_{j}(i) - u_j) = \sum_{k \neq j} \frac{u_k^\top (\bar{M}_i - \mathcal{G}) u_j}{\mu_j - \mu_k} v^\top u_k + O(\|\bar{M}_i - \mathcal{G}\|^2) \leq \sum_{k \neq j} \frac{2\gamma(i)}{\mu_j - \mu_k} v^\top u_k + O(\gamma(i)^2) \tag{12}$$

8
where we used $u_k^\top (\tilde{M}(i) - G) u_j \leq \|\tilde{M}(i) - G\|_2 \leq \bar{\gamma}(i) + \gamma(i) \leq 2\bar{\gamma}(i)$. Then for all $j$ such that $u_j \in V_\Delta$, \[
abla f_0(x) \nabla f_0(x) \nabla (\tilde{u}_j(i) - u_j) = \sum_{k \neq j} \frac{2\bar{\gamma}(i)}{\mu_j - \mu_k} \nabla f_0(x)^\top u_k + o(\gamma(i)^2) \leq \frac{2\bar{\gamma}(i)}{\Delta} \sqrt{d} \|\nabla f_0(x)\|_2 + o(\gamma(i)^2).\]

Note that the coefficients \[
\alpha_k = \frac{u_k^\top (\tilde{M}(i) - G) u_j}{\mu_j - \mu_k} + o(\gamma(i)^2) \quad k \neq j
\]
are a subset of coordinate values of vector $\tilde{u}_j(i) - u_j$ w.r.t. the orthonormal basis $u_1, \ldots, u_d$. Then, by Parseval’s identity, \[
4 \geq \|\tilde{u}_j(i) - u_j\|_2^2 \geq \sum_{k \neq j} \alpha_k^2.
\]
Therefore, it must be that \[
\max_{k \neq j} \left| \frac{u_k^\top (\tilde{M}(i) - G) u_j}{\mu_j - \mu_k} \right| \leq 2 + o(\gamma(i)^2).
\]

For any $j$ such that $u_j \in V_{\Delta}^1$, since $\mu_j - \mu_k \geq \Delta$ for all $u_k \in V_\Delta$, we may write \[
\nabla f_0(x) \nabla f_0(x) \nabla (\tilde{u}_j(i) - u_j) \leq \frac{2\bar{\gamma}(i)}{\Delta} \sum_{u_k \in V_\Delta} \nabla f_0(x)^\top u_k + \left(2 + o(\gamma(i)^2)\right) \sum_{u_k \in V_{\Delta}^1} \nabla f_0(x)^\top u_k + o(\gamma(i)^2) \leq \frac{2\bar{\gamma}(i)}{\Delta} \sqrt{d} \|P_{V_\Delta} \nabla f_0(x)\|_2 + \left(2 + o(\gamma(i)^2)\right) \sqrt{d} \|P_{V_{\Delta}^1} \nabla f_0(x)\|_2 + o(\gamma(i)^2)
\]
where $P_{V_\Delta}$ and $P_{V_{\Delta}^1}$ are the orthogonal projections onto, respectively, $V_\Delta$ and $V_{\Delta}^1$. Therefore, we have that \[
\|\nabla u_j f_0\|_\infty = \sup_{x \in \mathcal{X}} \nabla f_0(x)^\top (\tilde{u}_j(i) - u_j) 
\leq \sup_{x \in \mathcal{X}} \nabla f_0(x)^\top u_j + \sup_{x \in \mathcal{X}} \nabla f_0(x)^\top (\tilde{u}_j(i) - u_j) \leq \|\nabla u_j f_0\|_\infty + \frac{2\bar{\gamma}(i)}{\Delta} \sqrt{d} \|\nabla f_0\|_\infty + \left(2 + o(\gamma(i)^2)\right) \sqrt{d} \|\nabla f_0\|_\infty + o(\gamma(i)^2).
\]
(13)

Letting $\alpha_{\Delta}(i) = \frac{2\bar{\gamma}(i)}{\Delta} \sqrt{d} \|\nabla f_0\|_\infty + \left(2 + o(\gamma(i)^2)\right) \sqrt{d} \|\nabla f_0\|_\infty + o(\gamma(i)^2)$ we can upper bound (11) as follows \[
R_T(i + 1) \leq \left(8 \ln \left(e^{2i+1}\right) 12^{\rho_T} + 4 \left(\left(\mu_1 + 2\bar{\gamma}(i)\right) \sum_{j=1}^d \left(\|\nabla u_j f_0\|_\infty + \alpha_{\Delta}(i)\right)^2 / \mu_j\right)^2 2^{i+1} \frac{\bar{\gamma}(i)}{\Delta} \right). \]

Recall that, due to (10), the above holds at the end of each phase $i + 1$ with high probability. Now observe that $\bar{\gamma}(i) = \mathcal{O}(2^{-\alpha_1})$ and so $\alpha_{\Delta}(i) \leq (\|\nabla f_0\|_\infty / \Delta + \|\nabla f_0\|_\infty / \Delta)$. Hence, by summing over phases
\[ i = 1, \ldots, \lceil \log_2 T \rceil \] and applying the union bound,

\[
R_T = \sum_{i=1}^{\lceil \log_2 T \rceil} R_T(i)
\]

\[
\leq 8 \ln (eT) 12^d + \left( 1 + \sum_{j=1}^d \left( \frac{\| \nabla u_j f_0 \|_{\infty} + \| \nabla V_{\bot} f_0 \|_{\infty}}{\mu_j} \right)^2 \right)^{1/\bar{\rho}_T} T^{1+1/\bar{\rho}_T}
\]

concluding the proof. \qed

5 Conclusions and future work

We presented an efficient algorithm for online nonparametric regression which adapts to the directions along which the regression function \( f_0 \) is smoother. It does so by learning the Mahalanobis metric through the estimation of the gradient outer product matrix \( E[\nabla f_0(\mathbf{X})^\top \nabla f_0(\mathbf{X})] \). As a preliminary result, we analyzed the regret of a generalized version of the algorithm from [7], capturing situations where one competes against functions with directional Lipschitzness with respect to an arbitrary Mahalanobis metric. Our main result is then obtained through a phased algorithm that estimates the gradient outer product matrix while running online nonparametric regression on the same sequence. Both algorithms automatically adapt to the effective rank of the metric.

This work could be extended by investigating a variant of Algorithm 1 for classification, in which ball radii shrink at a nonuniform rate, depending on the mistakes accumulated within each ball rather than on time. This could lead to the ability of competing against functions \( f \) that are only locally Lipschitz. In addition, it is conceivable that under appropriate assumptions, a fraction of the balls could stop shrinking at a certain point when no more mistakes are made. This might yield better asymptotic bounds than those implied by Theorem 1, because \( \rho_T \) would never attain the ambient dimension \( d \).

Acknowledgments

Authors would like to thank Sébastien Gerchinovitz and Samory Kpotufe for useful discussions on this work. IK would like to thank Google for travel support. This work also was in parts funded by the European Research Council (ERC) under the European Union’s Horizon 2020 research and innovation programme (grant agreement no 637076).

References

[1] Romain Allez and Jean-Philippe Bouchaud. Eigenvector dynamics: general theory and some applications. Physical Review E, 86(4):046202, 2012.

[2] A. Bellet, A. Habrard, and M. Sebban. A Survey on Metric Learning for Feature Vectors and Structured Data. arXiv preprint arXiv:1306.6709, 2013.

[3] N. Cesa-Bianchi and G. Lugosi. Prediction, Learning, and Games. Cambridge University Press, 2006.

[4] X. Dong and D.-X. Zhou. Learning Gradients by a Gradient Descent Algorithm. Journal of Mathematical Analysis and Applications, 341(2):1018–1027, 2008.
[5] P. Gaillard and S. Gerchinovitz. A chaining Algorithm for Online Nonparametric Regression. In Conference on Learning Theory (COLT), 2015.

[6] Z.-C. Guo, Y. Ying, and D.-X. Zhou. Online Regularized Learning with Pairwise Loss Functions. Advances in Computational Mathematics, 43(1):127–150, 2017.

[7] E. Hazan and N. Megiddo. Online Learning with Prior Knowledge. In Learning Theory, pages 499–513. Springer, 2007.

[8] R. Jin, S. Wang, and Y. Zhou. Regularized Distance Metric Learning: Theory and Algorithm. In Conference on Neural Information Processing Systems (NIPS), 2009.

[9] S. Kpotufe, A. Boularias, T. Schultz, and K. Kim. Gradients Weights Improve Regression and Classification. Journal of Machine Learning Research, 17(22):1–34, 2016.

[10] S. Kpotufe and F. Orabona. Regression-Tree Tuning in a Streaming Setting. In Conference on Neural Information Processing Systems (NIPS), 2013.

[11] Robert Krauthgamer and James R Lee. Navigating nets: simple algorithms for proximity search. In Proceedings of the 15th annual ACM-SIAM Symposium on Discrete algorithms, pages 798–807. Society for Industrial and Applied Mathematics, 2004.

[12] S. Mukherjee and Q. Wu. Estimation of Gradients and Coordinate Covariation in Classification. Journal of Machine Learning Research, 7(Nov):2481–2514, 2006.

[13] S. Mukherjee and D.-X. Zhou. Learning Coordinate Covariances via Gradients. Journal of Machine Learning Research, 7(Mar):519–549, 2006.

[14] A. Rakhlin and K. Sridharan. Online Non-Parametric Regression. In Conference on Learning Theory (COLT), 2014.

[15] S. Trivedi, J. Wang, S. Kpotufe, and G. Shakhnarovich. A consistent Estimator of the Expected Gradient Outerproduct. In Conference on Uncertainty in Artificial Intelligence (UAI), 2014.

[16] V. Vovk. On-line regression competitive with reproducing kernel Hilbert spaces. In International Conference on Theory and Applications of Models of Computation. Springer, 2006.

[17] Vladimir Vovk. Metric entropy in competitive on-line prediction. arXiv preprint cs/0609045, 2006.

[18] Vladimir Vovk. Competing with wild prediction rules. Machine Learning, 69(2):193–212, 2007.

[19] M. Wainwright. High-Dimensional Statistics: A Non-Asymptotic Viewpoint (In Preparation). 2017.

[20] Y. Wang, R. Khardon, D. Pechyony, and R. Jones. Generalization Bounds for Online Learning Algorithms with Pairwise Loss Functions. In Conference on Learning Theory (COLT), 2012.

[21] K. Q. Weinberger and L. K. Saul. Distance Metric Learning for Large Margin Nearest Neighbor Classification. Journal of Machine Learning Research, 10:207–244, 2009.

[22] J. H. Wilkinson. The Algebraic Eigenvalue Problem, volume 87. Clarendon Press Oxford, 1965.

[23] Qiang Wu, Justin Guinney, Mauro Maggioni, and Sayan Mukherjee. Learning gradients: predictive models that infer geometry and statistical dependence. Journal of Machine Learning Research, 11(Aug):2175–2198, 2010.
A Nonparametric gradient learning

In this section we describe a nonparametric gradient learning algorithm introduced in [15]. Throughout this section, we assume instances \( x_t \) are realizations of i.i.d. random variables \( X_t \) drawn according to some fixed and unknown distribution \( \mu \) which has a continuous density on its support \( \mathcal{X} \). Labels \( y_t \) are generated according to the noise model \( y_t = f(x_t) + \nu(x_t) \), where \( \nu(x) \) is a subgaussian zero-mean random variable for all \( x \in \mathcal{X} \). The algorithm computes a sequence of estimates \( \hat{f}_1, \hat{f}_2, \ldots \) of the regression function \( f \) through kernel regression. Let \( \mathcal{X}_n \equiv \{x_1, \ldots, x_n\} \subseteq \mathcal{X} \) be the data observed so far and let \( y_1, \ldots, y_n \) their corresponding labels. Let \( K : \mathbb{R}_+ \to \mathbb{R}_+ \) be a nonincreasing kernel, strictly positive on \([0, 1)\), and such that \( K(1) = 0 \). Then the estimate at time \( n \) is defined by

\[
\hat{f}_n(x) = \sum_{t=1}^{n} y_t \omega_t(x) \quad \text{where} \quad \omega_t(x) = \begin{cases} K\left( \frac{\|x - x_t\|}{\varepsilon_n} \right) & \text{if } B(x, \varepsilon_n) \cap \mathcal{X}_n \neq \emptyset, \\ \frac{1}{n} & \text{otherwise} \end{cases}
\]

where \( \varepsilon_n > 0 \) is the kernel scaling parameter. We then approximate the gradient of \( \hat{f} \) at any given point through the finite difference method

\[
\Delta_i(x) = \frac{1}{2\tau_n} \left( \hat{f}(x + \tau_ne_i) - \hat{f}(x - \tau_ne_i) \right) \quad \text{for } i = 1, \ldots, d
\]

where \( \tau_n > 0 \) is a parameter. Let further

\[
A_i(x) = \mathbb{I} \left\{ \min_{b \in \{-\tau_n, \tau_n\}} \mu_n(B(x + be_i, \varepsilon/2)) \geq \frac{2d}{n} (\ln 2n) \right\} \quad \text{for } i = 1, \ldots, d
\]

where \( \mu_n \) is the empirical distribution of \( \mu \) after observing \( \mathcal{X}_n \), and define the gradient estimate

\[
\hat{\nabla} f(x_t) = \left( \Delta_1(x_t)A_1(x_t), \ldots, \Delta_d(x_t)A_d(x_t) \right).
\]

The algorithm outputs at time \( n \) the gradient outer product estimate

\[
\hat{G}_n = \frac{1}{n} \sum_{t=1}^{n} \hat{\nabla} f(x_t)\hat{\nabla} f(x_t)\top
\]

Let \( G = \mathbb{E} \left[ \nabla f(X)\nabla f(X)\top \right] \) be the expected gradient outer product, where \( X \) has law \( \mu \). The next lemma states that, under Assumption 1, \( \hat{G}_n \) is a consistent estimate of \( G \).

**Lemma (Consistency of the Expected Gradient Outerproduct Estimator [15, Theorem 1]).** If Assumption 1 holds, then there exists a nonnegative and nonincreasing sequence \( \{\gamma_n\}_{n \geq 1} \) such that for all \( n \), the estimated gradient outer product (7) computed with parameters \( \varepsilon_n > 0 \), and \( 0 < \tau_n < \tau_0 \) satisfies

\[
\|\hat{G}_n - G\|_2 \leq \gamma_n \text{ with high probability with respect to the random draw of } X_1, \ldots, X_n.
\]

Moreover, if \( \tau_n = \Theta\left( \varepsilon_n^{1/2} \right) \), \( \varepsilon_n = \Omega\left( (\ln n)^{\frac{3}{4}d^{\frac{1}{4}}} n^{-\frac{1}{2}} \right) \), and \( \varepsilon_n = O\left( n^{-\frac{1}{2(d+1)}} \right) \) then \( \gamma_n \to 0 \) as \( n \to \infty \).

The actual rate of convergence depends, in a complicated way, on parameters related to the distribution \( \mu \) and the regression function \( f \). In our application of Lemma 4 we assume \( \gamma_n \lesssim n^{-\alpha} \) for all \( n \) large enough and for some \( \alpha > 0 \). Note also that the convergence of \( \hat{G}_n \) to \( G \) holds in probability with respect to the random draw of \( X_1, \ldots, X_n \). Hence there is a confidence parameter \( \delta \) which is not shown here. However, the dependence of the convergence rate on \( \frac{1}{\delta} \) is only polylogarithmic and therefore not problematic for our applications.
B  Proofs from Section 3

Lemma (Volumetric packing bound). Consider a pair of norms \(\|\cdot\|, \|\cdot\|'\) and let \(B, B' \subset \mathbb{R}^d\) be the corresponding unit balls. Then

\[
\mathcal{M}(B, \varepsilon, \|\cdot\|') \leq \frac{\text{vol}(B + \frac{\varepsilon}{2}B')}{\text{vol}(\frac{\varepsilon}{2}B')}
\]

Proof. Let \(\{x_1, \ldots, x_M\}\) be a maximal \(\varepsilon\)-packing of \(B\) according to \(\|\cdot\|'\). Since we have a packing, the \(\|\cdot\|'\)-balls of radius \(\varepsilon/2\) and centers \(x_1, \ldots, x_M\) are disjoint, and their union is contained in \(B + \frac{\varepsilon}{2}B'\). Thus,

\[
\mathcal{M} \text{vol}(\frac{\varepsilon}{2}B') \leq \text{vol}(B + \frac{\varepsilon}{2}B')
\]

which concludes the proof.

Lemma (Ellipsoid packing bound). If \(B\) is the unit Euclidean ball then

\[
\mathcal{M}(B, \varepsilon, \|\cdot\|_M) \leq \left(\frac{8\sqrt{2}}{\varepsilon}\right)^s \prod_{i=1}^s \sqrt{\lambda_i} \quad \text{where} \quad s = \max \left\{ i : \sqrt{\lambda_i} \geq \varepsilon, i = 1, \ldots, d \right\}.
\]

Proof. The change of variable \(x' = M^{1/2}x\) implies \(\|x\|_2 = \|x'\|_{M^{-1}}\) and \(\|x\|_M = \|x'\|_2\). Therefore \(\mathcal{M}(B, \varepsilon, \|\cdot\|_M) = \mathcal{M}(E, \varepsilon, \|\cdot\|_2)\) where \(E = \{x \in \mathbb{R}^d : \|x\|_{M^{-1}} \leq 1\}\) is the unit ball in the norm \(\|\cdot\|_{M^{-1}}\). Next, we write the coordinates \((x_1, \ldots, x_d)\) of any point \(x \in \mathbb{R}^d\) using the orthonormal basis \(u_1, \ldots, u_d\). Consider the truncated ellipsoid \(\tilde{E} = \{x \in E : x_i = 0, i = s + 1, \ldots, d\}\). By adapting an argument from [19], we prove that any \(\varepsilon\)-cover of \(\tilde{E}\) according to \(\|\cdot\|_2\) is also a \((\varepsilon\sqrt{2})\)-cover of \(E\) according to the same norm. Indeed, let \(\tilde{S} \subset \tilde{E}\) be a \(\varepsilon\)-cover of \(\tilde{E}\). Fix any \(x \in E\) and let

\[
\min_{\tilde{x} \in \tilde{S}} \|x - \tilde{x}\|_2 = \min_{\tilde{x} \in \tilde{S}} \sum_{j=1}^s (x_j - \tilde{x}_j)^2 + \sum_{j=s+1}^d x_j^2
\]

\[
\leq \varepsilon^2 + \sum_{j=s+1}^d x_j^2 \quad \text{(since \(\tilde{S}\) is a \(\varepsilon\)-covering of \(\tilde{E}\))}
\]

\[
\leq \varepsilon^2 + \lambda_{s+1} \sum_{j=s+1}^d \frac{x_j^2}{\lambda_j} \quad \text{(since \(\lambda_{s+1}/\lambda_j \geq 1\) for } j = s + 1, \ldots, d\)}
\]

\[
\leq 2\varepsilon^2
\]

where the last inequality holds since \(\lambda_{s+1} \leq \varepsilon^2\) and since \(\|x\|_{M^{-1}}^2 = \sum_{i=1}^d x_i^2/\lambda_i \leq 1\) for any \(x \in E\), where \(x_i = u_i^T x\) for all \(i = 1, \ldots, d\). Let \(B' \subset \mathbb{R}^d\) be the unit Euclidean ball, and let \(\tilde{B}' = \{x \in B' : x_i = 0, i = s + 1, \ldots, d\}\) be its truncated version. Since \(\lambda_i \geq \varepsilon^2\) for \(i = 1, \ldots, s\) we have that for all \(x \in \varepsilon\tilde{B}'\), \(x_1^2 + \cdots + x_s^2 \leq \varepsilon^2\) and so

\[
\|x\|_{M^{-1}}^2 = \sum_{i=1}^s \frac{x_i^2}{\lambda_i} \leq \sum_{i=1}^s \frac{\varepsilon^2}{\lambda_i} \leq 1
\]
Therefore \( \varepsilon \tilde{B}' \subseteq \tilde{E} \) which implies \( \text{vol}(\tilde{E} + \varepsilon \frac{\tilde{B}'}{2}) \leq \text{vol}(2 \tilde{E}) \).

\[
\mathcal{M}(E, 2\varepsilon \sqrt{2}, \| \cdot \|_2) \leq \mathcal{N}(E, \varepsilon \sqrt{2}, \| \cdot \|_2) \\
\leq \mathcal{N}(\tilde{E}, \varepsilon, \| \cdot \|_2) \\
\leq \mathcal{M}(\tilde{E}, \varepsilon, \| \cdot \|_2) \\
\leq \frac{\text{vol}(\tilde{E} + \varepsilon \frac{\tilde{B}'}{2})}{\text{vol}(\frac{\tilde{B}'}{2})} \\
\leq \frac{\text{vol}(2 \tilde{E})}{\text{vol}(\frac{\tilde{B}'}{2})} = \left( \frac{4}{\varepsilon} \right)^s \frac{\text{vol}(\tilde{E})}{\text{vol}(\tilde{B}')} \\
\text{(by Lemma 1)}
\]

Now, using the standard formula for the volume of an ellipsoid,

\[
\text{vol}(\tilde{E}) = \text{vol}(\tilde{B}') \prod_{i=1}^s \sqrt{\lambda_i}.
\]

This concludes the proof. \( \square \)

The following lemma states that whenever \( f \) has bounded partial derivatives with respect to the eigenbase of \( M \), then \( f \) is Lipschitz with respect to \( \| \cdot \|_M \).

**Lemma** (Bounded derivatives imply Lipschitzness in \( M \)-metric). Let \( f : X \to \mathbb{R} \) be everywhere differentiable. Then for any \( x, x' \in X \),

\[
|f(x) - f(x')| \leq \|x - x'\|_M \sqrt{\sum_{i=1}^d \frac{\|\nabla u_i f\|_\infty^2}{\lambda_i}}.
\]

**Proof.** By the mean value theorem, there exists a \( z \) on the segment joining \( x \) and \( y \) such that \( f(x) - f(y) = \nabla f(z)^\top (x - y) \). Hence

\[
f(x) - f(y) = \nabla f(z)^\top (x - y) \\
= \sum_{i=1}^d \nabla f(z)^\top u_i u_i^\top (x - y) \\
\leq \sum_{i=1}^d \left( \sup_{z' \in X} \nabla f(z')^\top u_i \right) u_i^\top (x - y) \\
= \sum_{i=1}^d \frac{\|\nabla u_i f\|_\infty}{\sqrt{\lambda_i}} \left( \sqrt{\lambda_i} u_i^\top (x - y) \right) \\
\leq \sqrt{\sum_{i=1}^d \frac{\|\nabla u_i f\|_\infty^2}{\lambda_i}} \sqrt{\sum_{i=1}^d \lambda_i (u_i^\top (x - y))^2} \text{ (by the Cauchy-Schwarz inequality)} \\
= \|x - y\|_M \sqrt{\sum_{i=1}^d \frac{\|\nabla u_i f\|_\infty^2}{\lambda_i}}.
\]

By symmetry, we can upper bound \( f(y) - f(x) \) with the same quantity. \( \square \)

Now we are ready to prove the regret bound.
We proceed by decomposing the regret into a local (estimation) and a global (approximation) term, \( \kappa \), where

**Theorem (Regret with Fixed Metric).** Suppose Algorithm 1 is run with a positive definite matrix \( M \) with eigenbasis \( \{u_1, \ldots, u_d\} \) and eigenvalues \( 1 = \lambda_1 \geq \cdots \geq \lambda_d > 0 \). Then, for any differentiable \( f: \mathcal{X} \to \mathcal{Y} \) we have that

\[
R_T(f) \leq \sum_{t=1}^T \frac{\sqrt{\det(\kappa(M))} + \sqrt{\sum_{i=1}^d \|\nabla u_i, f\|_\infty^2 \lambda_i}}{T^{\rho_T + \rho_T}}
\]

where \( \kappa = (\rho_T, T) \leq \rho_T \leq d \).

**Proof.** Let \( S_t \) be the value of the variable \( S \) at the end of time \( t \). Hence \( S_0 = \emptyset \). The functions \( \pi_t: \mathcal{X} \to \{1, \ldots, t\} \) for \( t = 1, 2, \ldots \) map each data point \( x \) to its closest (in norm \( \| \cdot \|_M \)) center in \( S_{t-1} \).

\[
\pi_t(x) = \begin{cases} 
\arg\min_{s \in S_{t-1}} \|x - x_s\|_M & \text{if } S_{t-1} \neq \emptyset \\
\text{otherwise.} & 
\end{cases}
\]

The set \( T_s \) contain all data points \( x_t \) that at time \( t \) belonged to the ball with center \( x_s \) and radius \( \varepsilon_t \),

\[
T_s = \{ t : \|x_t - x_s\|_M \leq \varepsilon_t, \ t = s, \ldots, T \}.
\]

Finally, \( y_s^* \) is the best fixed prediction for all examples \((x_t, y_t)\) such that \( t \in T_s \),

\[
y_s^* = \arg\min_{y \in \mathcal{Y}} \sum_{t \in T_s} \ell_t(y) = \frac{1}{|T_s|} \sum_{t \in T_s} y_t.
\]

We proceed by decomposing the regret into a local (estimation) and a global (approximation) term,

\[
R_T(f) = \sum_{t=1}^T \left( \ell_t(\hat{y}_t) - \ell_t(f(x_t)) \right) = \sum_{t=1}^T \left( \ell_t(\hat{y}_t) - \ell_t(y^*_{\pi_t(x_t)}) \right) + \sum_{t=1}^T \left( \ell_t(y^*_{\pi_t(x_t)}) - \ell_t(f(x_t)) \right).
\]

The estimation term is bounded as

\[
\sum_{t=1}^T \left( \ell_t(\hat{y}_t) - \ell_t(y^*_{\pi_t(x_t)}) \right) = \sum_{s \in S_T} \sum_{t \in T_s} \left( \ell_t(\hat{y}_t) - \ell_t(y^*_s) \right) \leq 8 \sum_{s \in S_T} \ln(\varepsilon|N_s|) \leq 8 \ln(\varepsilon T)|S_T|.
\]

The first inequality is a known bound on the regret under square loss \( [3, \text{page 43}] \). We upper bound the size of the final packing \( S_T \) using Lemma 2,

\[
|S_T| \leq M(B, \varepsilon_T, \| \cdot \|_M) \leq \left( \frac{8\sqrt{2}}{\varepsilon_T} \right)^{\kappa} \prod_{i=1}^\infty \sqrt{\lambda_i} \leq (8\sqrt{2})^{\kappa} \sqrt{\det(\kappa(M))} T^{\rho_T + \rho_T}
\]

where \( \kappa = (\rho_T, T) \). Therefore, since \( \rho_T \geq (\rho_T, T) \),

\[
\sum_{t=1}^T \left( \ell_t(\hat{y}_t) - \ell_t(y^*_{\pi_t(x_t)}) \right) \leq 8 \ln(\varepsilon T)(8\sqrt{2})^{\rho_T} \sqrt{\det(\kappa(M))} T^{\rho_T + \rho_T}.
\]

Next, we bound the approximation term. Using (14) we have

\[
\sum_{t=1}^T \left( \ell_t(y^*_{\pi_t(x_t)}) - \ell_t(f(x_t)) \right) \leq \sum_{t=1}^T \left( \ell_t(f(x_{\pi_t(x_t)})) - \ell_t(f(x_t)) \right).
\]
Note that $\ell_t$ is 2-Lipschitz because $y_t, \hat{y}_t \in [0, 1]$. Hence, using Lemma 3,
\[
\ell_t(f(x_{\pi_t}(x_t))) - \ell_t(f(x_t)) \leq 2\|x_t - x_{\pi_t}(x_t)\|_M \sqrt{\sum_{i=1}^d \frac{\|\nabla_u f\|^2}{\lambda_i}}
\]
\[
\leq 2 \varepsilon_t \sqrt{\sum_{i=1}^d \frac{\|\nabla_u f\|^2}{\lambda_i}}.
\]

Recalling $\varepsilon_t = t^{-\frac{1}{1+\rho_t}}$ where $\rho_t \leq \rho_t+1$, we write
\[
\sum_{t=1}^T t^{-\frac{1}{1+\rho_t}} \leq \sum_{t=1}^T t^{-\frac{1}{1+\rho_T}} \leq \int_0^T t^{-\frac{1}{1+\rho_T}} \, dt = \left(1 + \frac{1}{\rho_T}\right) T^{\rho_T} \leq 2 T^{\rho_T}.
\]

Thus we may write
\[
\sum_{t=1}^T \left(\ell_t(y^*_t(x_t)) - \ell_t(f(x_t))\right) \leq 4 \left(\sqrt{\sum_{i=1}^d \frac{\|\nabla_u f\|^2}{\lambda_i}}\right) T^{\rho_T}.
\]

The proof is concluded by combining the above with (15).

\[\square\]

C Proofs from Section 4

Lemma 6. Let $\mu, \alpha > 0$ and $d \geq 1$. Then the derivative of
\[
F(t) = \left(\mu + 2(T_0 + t)^{-\alpha}\right) t^{\frac{2}{1+\alpha}}
\]

is positive for all $t \geq 1$ when
\[
T_0 \geq \left(\frac{d+1}{2\mu}\right)^{1/\alpha}.
\]

Proof. We have that $F'(t) \geq 0$ if and only if
\[
t \leq \frac{2(T_0 + t)}{\alpha(d + 1)}(1 + (T_0 + t)^\alpha \mu_d)
\]

This is implied by
\[
t \leq \frac{2\mu_d(T_0 + t)^{1+\alpha}}{\alpha(d + 1)}\quad \text{or, equivalently,} \quad T_0 \geq A^{1/(1+\alpha)} t^{1/(1+\alpha)} - t
\]

where $A = \alpha(d + 1)/(2\mu_d)$. The right-hand side $A^{1/(1+\alpha)} t^{1/(1+\alpha)} - t$ is a concave function of $t$. Hence the maximum is found at the value of $t$ where the derivative is zero, this value satisfies
\[
A^{1/(1+\alpha)} t^{-\alpha/(1+\alpha)} = 1 \quad \text{which solved for } t \text{ gives } t = A^{1/\alpha}(1 + \alpha)^{-(1+\alpha)/\alpha}.
\]

Substituting this value of $t$ in $A^{1/(1+\alpha)} t^{1/(1+\alpha)} - t$ gives the condition $T_0 \geq A^{1/\alpha}(1 + \alpha)^{-(1+\alpha)/\alpha}$ which is satisfied when $T_0 \geq A^{1/\alpha}$. \[\square\]