Algorithmic Foundations of Learning **Lecture 10**

Non-Euclidean Settings. Mirror Descent Methods

Lecturer: Patrick Rebeschini *Version: December 8, 2021*

10.1 Introduction

Last time we introduced a general class of algorithms known as gradient methods and discussed their convergence guarantees in a variety of settings in convex optimization. We applied this analysis to the case of linear predictors with parameters constrained in the Euclidean ball, and showed that in the case of a Lipschitz loss function the rate of convergence of the optimization error (with respect to iteration time t) matches (modulo universal constants) the rate of convergence of the statistical error (with respect to the amount of training data n). This, in turns, suggests a principled approach to stop the optimization routine, as originally put forward in [1]: run the projected subgradient method for $t \sim n$ time steps.

The main idea behind this argument is a natural one: as the learning problem has a certain level of intrinsic noise (note that the training data is modeled as random variables, hence noisy by definition), there is no point in solving the empirical risk minimization problem with an accuracy that is below the level of the noise, as that would be a waste of computational resources. This is one of the main differences that set optimization for machine learning apart from optimization for deterministic settings. At the same time, the analysis that we presented is only based on upper bounds, so one should be careful about drawing conclusions from it!

Today, we consider the non-Euclidean setting, and show that in this case the same phenomenon does not occur for gradient methods: the upper bound that we can derive for the optimization error of the subgradient method with the simplex constraint (analogously, the ℓ_1-ball) yields a rate of convergence that is slower (with respect to the dimension d of the problem) as compared to the statistical rate we previously derived for the same problem in Lecture 3. This mismatch will prompt us to develop a more general class of algorithms know as *mirror descent* methods.

We briefly recall our findings in the Euclidean setting.

10.1.1 Euclidean Settings

Let $(X_1, Y_1), \dots, (X_n, Y_n) \in \mathcal{X} \times \{-1, 1\}$ be the training data, with $\mathcal{X} \subseteq \mathbb{R}^d$. Let $\varphi : \mathbb{R} \to \mathbb{R}_+$ be a given convex loss function. Let

$$\mathcal{A}_2 = \{x \in \mathbb{R}^d \to a(x) = w^\top x : w \in \mathcal{W}_2\},$$

where $\mathcal{W}_2 = \{w \in \mathbb{R}^d : \|w\|_2 \leq c_2^{\mathcal{W}}\}$. Let us assume that the loss function φ is γ_φ-Lipschitz and let $c_2^{\mathcal{X}} := \max_{x \in \mathcal{X}} \|x\|_2$.

The risk minimization problem reads

$$\begin{aligned} \underset{w}{\text{minimize}} \quad & r(w) = \mathbf{E}\varphi(w^\top XY) \\ \text{subject to} \quad & w \in \mathcal{W}_2 \end{aligned}$$

Let $w_2^\star$ be a minimizer of this problem. The empirical risk minimization problem reads

$$\begin{aligned} \underset{w}{\text{minimize}} \quad & R(w) = \frac{1}{n}\sum_{i=1}^{n} \varphi(w^\top X_i Y_i) \\ \text{subject to} \quad & w \in \mathcal{W}_2 \end{aligned}$$

Let $W_2^\star$ be a minimizer of this problem.

Recall the error decomposition derived in Section 1.2.2, which holds in particular for the output of the projected subgradient method $\overline{W}_t = \frac{1}{t}\sum_{s=1}^{t} W_s$ at time t applied to the empirical risk minimization problem:

$$r(\overline{W}_t) - r(w_2^\star) \le \underbrace{R(\overline{W}_t) - R(W_2^\star)}_{\texttt{Optimization}_2} + \underbrace{\sup_{w\in\mathcal{W}_2}\{r(w) - R(w)\} + \sup_{w\in\mathcal{W}_2}\{R(w) - r(w)\}}_{\texttt{Statistics}_2}.$$

The expected value of the $\texttt{Statistics}_2$ term can be bounded using the tools developed in the first part of this course (see Proposition 3.2, in particular). We have

$$\boxed{\mathbf{E}\,\texttt{Statistics}_2 \le \frac{4c_2^{\mathcal{X}} c_2^{\mathcal{W}} \gamma_\varphi}{\sqrt{n}}}$$

The $\texttt{Optimization}_2$ term can be bounded by Theorem 9.3, giving

$$\boxed{\texttt{Optimization}_2 \le \frac{2c_2^{\mathcal{X}} c_2^{\mathcal{W}} \gamma_\varphi}{\sqrt{t}}}$$

In this example there is a perfect matching (modulo universal constants) between the statistical rate and the optimization rate, so that running the algorithm for $t \sim n$ time steps suffices for the upper bound on the optimization error to be of the same order as the upper bound on the statistical error. This argument gives a principled approach to obtain *computational* savings in the training phase, as we can reliably stop the optimization algorithm after $t \sim n$ time steps.

10.1.2 Non-Euclidean Settings

We are now interested in developing the same analysis in the case when we replace $\mathcal{A}_2$ by the family of predictors with parameters contained in the d-dimensional simplex, namely,

$$\mathcal{A}_\Delta = \{x \in \mathbb{R}^d \to a(x) = w^\top x : w \in \Delta_d\},$$

where $\Delta_d := \{w = (w_1, \dots, w_d) \in \mathbb{R}^d : \|w\|_1 = 1, w_1, \dots, w_d \ge 0\}$.

The risk minimization problem reads

$$\begin{aligned} \underset{w}{\text{minimize}} \quad & r(w) = \mathbf{E}\varphi(w^\top XY) \\ \text{subject to} \quad & w \in \Delta_d \end{aligned}$$

Let $w_\Delta^\star$ be a minimizer of this problem. The empirical risk minimization problem reads

$$\begin{aligned} \underset{w}{\text{minimize}} \quad & R(w) = \frac{1}{n}\sum_{i=1}^{n} \varphi(w^\top X_i Y_i) \\ \text{subject to} \quad & w \in \Delta_d \end{aligned} \tag{10.1}$$

Let $W_\Delta^\star$ be a minimizer of this problem. The error decomposition reads

$$r(\overline{W}_t) - r(w_\Delta^\star) \le \underbrace{R(\overline{W}_t) - R(W_\Delta^\star)}_{\texttt{Optimization}_\Delta} + \underbrace{\sup_{w\in\Delta_d}\{r(w) - R(w)\} + \sup_{w\in\Delta_d}\{R(w) - r(w)\}}_{\texttt{Statistics}_\Delta}.$$

Also in this case the expected value of the $\texttt{Statistics}_\Delta$ term can be bounded using the tools developed in the first part of this course (see Proposition 3.4, in particular). We find

$$\begin{aligned}\mathbf{E}\sup_{w\in\Delta_d}\{r(w) - R(w)\} &\le 2\,\mathbf{E}\,\texttt{Rad}(\mathcal{L}_\Delta \circ \{Z_1,\dots,Z_n\}) \le 2\gamma_\varphi \mathbf{E}\,\texttt{Rad}(\mathcal{A}_\Delta \circ \{X_1,\dots,X_n\}) \\ &\le 2\gamma_\varphi c_1^{\mathcal{W}} \mathbf{E}\,\frac{\max_i \|X_i\|_\infty}{\sqrt{n}}\sqrt{2\log d} \\ &\le \frac{2c_\infty^{\mathcal{X}} c_1^{\mathcal{W}}\gamma_\varphi}{\sqrt{n}}\sqrt{2\log d},\end{aligned}$$

where $c_\infty^{\mathcal{X}} := \max_{x\in\mathcal{X}}\|x\|_\infty$ and $c_1^{\mathcal{X}} = 1$. Hence,

$$\boxed{\mathbf{E}\,\texttt{Statistics}_\Delta \le 4c_\infty^{\mathcal{X}} c_1^{\mathcal{W}}\gamma_\varphi\sqrt{\frac{2\log d}{n}}} \tag{10.2}$$

Let us now investigate what happens to the optimization error when we apply the projected subgradient method to this problem. Note that the empirical risk R is $(\sqrt{d}c_\infty^{\mathcal{X}}\gamma_\varphi)$-Lipschitz, as

$$\begin{aligned}|R(w) - R(u)| &\le \frac{1}{n}\sum_{i=1}^n |\varphi(w^\top X_i Y_i) - \varphi(u^\top X_i Y_i)| \le \frac{\gamma_\varphi}{n}\sum_{i=1}^n |Y_i(w-u)^\top X_i| \le \gamma_\varphi \|w-u\|_2 \max_{i\in[n]}\|X_i\|_2 \\ &\le \sqrt{d}c_\infty^{\mathcal{X}}\gamma_\varphi\|w-u\|_2,\end{aligned}$$

where we used that $|Y_i| = 1$, the Cauchy-Schwarz's inequality, and the fact that $\|x\|_2 \le \sqrt{d}\|x\|_\infty$ (note that this inequality is sharp: just consider the case of the all-ones vector). Therefore, by applying the projected subgradient method to problem (10.1) we have, by Theorem 9.3,

$$\texttt{Optimization}_\Delta \le \frac{2c_\infty^{\mathcal{X}} c_1^{\mathcal{W}}\gamma_\varphi}{\sqrt{t}}\sqrt{d}, \tag{10.3}$$

where we used that $\|W_1 - W^\star\|_2 \le \|W_1 - W^\star\|_1 \le \|W_1\|_1 + \|W^\star\|_1 \le 2c_1^{\mathcal{W}}$.

Remark 10.1 *This example shows that gradient methods only provide dimension-free results if the objective function and the constraints set "behave well" with respect to the Euclidean geometry, as previously explained in Remark 9.6. In the present case, the Lipschitz constant of the empirical risk R explicitly depends on the dimension d, so the final rate is dimension-dependent.*

While the expected value of the `Statistics` term is guaranteed to grow at most logarithmically with the dimension d, the convergence rate of the subgradient method is only guaranteed to grow at most polynomially with d. This is an example where one would like to apply gradient methods on a non-Euclidean geometry: the probability simplex Δ_d. However, gradient methods are designed for the Euclidean geometry! Recall, in fact, that all the definitions we gave for α-strong convexity, β-smoothness, and γ-Lipschitz continuity, as well as the bounds for the constraint set $\mathcal{C}$, are expressed in terms of the Euclidean norm $\|\cdot\|_2$.

Overcoming the dependence of gradient methods on the Euclidean geometry prompts for the design of a more general class of algorithms that can adapt to the geometry of the problem at hand. This is achieved by the class of algorithms that we define next, known as mirror descent methods. These algorithms will allow us to solve the empirical risk minimization problem (10.1) with a bound on the optimization term that scales logarithmically with d, matching the statistical guarantees previously derived.

Remark 10.2 (Majority Votes or Boosting) *The non-Euclidean setting that we have introduced is a popular one in machine learning, as it refers to the case of Boosting, a meta-learning algorithm that assembles given predictors to improve their learning capabilities. Given the training data* $(X_1, Y_1), \dots, (X_n, Y_n) \in \mathcal{X} \times \{-1, 1\}$*, where now* $\mathcal{X} \subset \mathbb{R}^m$*, assume that we are given a family of* d *base classifiers* $h_1, \dots, h_d : \mathbb{R}^m \to \mathbb{R}$*. For each* $x \in \mathcal{X}$*, let* $h(x) = (h_1(x), \dots, h_d(x)) \in \mathbb{R}^d$ *be the vector that encodes the prediction of the* d *base classifiers on the given data point* x*. In this case, for any* $w \in \Delta_d$*,* $w^\top h(x)$ *represents a convex combination of the base classifiers, and the risk minimization problems read*

$$\begin{aligned} \underset{w}{\text{minimize}} \quad & r(w) = \mathbf{E}\varphi(w^\top h(x)Y) \\ \text{subject to} \quad & w \in \Delta_d \end{aligned}$$

$$\begin{aligned} \underset{w}{\text{minimize}} \quad & R(w) = \frac{1}{n}\sum_{i=1}^{n} \varphi(w^\top h(x_i)Y_i) \\ \text{subject to} \quad & w \in \Delta_d \end{aligned}$$

which corresponds to the problems defined above with the substitutions $x \to h(x)$*.*

10.2 Mirror Descent: Setup

From the boosting example it is clear that we would like an algorithm that works in *any* geometry, not just in the Euclidean one. To achieve this, we need to measure distances not with respect to the Euclidean norm $\|\cdot\|_2$, but with respect to any generic norm $\|\cdot\|$. This is the idea of the *mirror descent* algorithm.

We can immediately state the equivalent of the local-to-global geometrical properties that are typically used to establish rate of convergence for algorithms with respect to a generic norm $\|\cdot\|$.

- α**-Strongly convex:** There exits $\alpha > 0$ such that $f(x) - f(y) \le \nabla f(x)^\top (x - y) - \frac{\alpha}{2}\|x - y\|^2 \ \forall x, y \in \mathcal{C}$.
- β**-Smooth:** There exits $\beta > 0$ such that $f(x) - f(y) \ge \nabla f(x)^\top (x - y) - \frac{\beta}{2}\|x - y\|^2$ for any $x, y \in \mathcal{C}$.
- γ**-Lipschitz:** There exits $\gamma > 0$ such that $|f(x) - f(y)| \le \gamma\|x - y\|$ for any $x, y \in \mathcal{C}$.

However, replacing the norm in gradient descent is not entirely straightforward. The projected gradient descent algorithm works in any arbitrary Hilbert space, where the norm of vectors is associated with an inner product. Now, suppose we are interested in optimization on a Banach space $\mathcal{D}$ where the norm does not derive from an inner product. In this case, gradient descent does not even make sense since the gradient $\nabla f(x)$ is an element of the dual space, thus the term $x - \nabla f(x)$ is not even defined.

To see this, we briefly review some basics about derivatives on normed vector spaces. Denote $\mathcal{D}$ a normed vector space and $\|\cdot\|_{\mathcal{D}}$ an associated norm. The *dual space* $\mathcal{D}^*$ of $\mathcal{D}$ is defined as the vector space of all linear maps $d^* : \mathcal{D} \to \mathbb{R}$, equipped with the *operator* or *dual norm*

$$\|d^*\|_{\mathcal{D}^*} := \sup\{|d^*(d)|; d \in \mathcal{D}, \|d\|_{\mathcal{D}} = 1\}.$$

Remark 10.3 *If* $\mathcal{D} = \mathbb{R}^d$ *and* $\|\cdot\|$ *is a given vector norm, then linear functionals can be written as a scalar product, i.e., we have* $d^*(x) = g^\top x$ *for some* $g \in \mathbb{R}^d$*, and the dual norm reads*

$$\|g\|_* := \|d^*\|_{\mathcal{D}^*} := \sup\left\{|g^\top x|; x \in \mathbb{R}^d, \|x\| = 1\right\}.$$

We note that the Lipschitz property with respect to a norm $\|\cdot\|$ can be phrased as the property that there exists $\gamma > 0$ such that for any $x \in \mathbb{R}^d$, any subgradient $g \in \partial f(x)$ satisfies $\|g\|_* \leq \gamma$.

The Fréchet derivative is a derivative defined on Banach spaces.

Definition 10.4 (Fréchet derivative) *Given normed spaces $(\mathcal{D}, \|\cdot\|_{\mathcal{D}})$ and $(\mathcal{D}', \|\cdot\|_{\mathcal{D}'})$, we say that an operator $T\colon \mathcal{C} \subset \mathcal{D} \to \mathcal{D}'$ (where $\mathcal{C}$ is open) is Fréchet differentiable at $d \in \mathcal{C}$ if there exists a bounded linear operator $D_d T\colon \mathcal{D} \to \mathcal{D}'$, such that*

$$\lim_{h \to 0} \frac{\|T(d+h) - T(d) - D_d T(h)\|_{\mathcal{D}'}}{\|h\|_{\mathcal{D}}} = 0.$$

The operator $D_d T$ is called the Fréchet derivative of T at the point d.

If $\mathcal{D}' = \mathbb{R}$, then $D_d T$ is an element of the dual $\mathcal{D}^*$.

Going back to our gradient descent algorithm this means that in general the equation $x - \eta \nabla f(x)$ does not make sense, because $\nabla f(x)$ is an element of the dual and x is an element of the primal. The exception in case of the Euclidean norm arises because the Riesz representation theorem implies that the dual of a Hilbert space is isometric to its primal.

Mirror descent allows us to circumvent this problem by mapping the current point of our descent algorithm to its dual, perform the gradient descent step and map back to our primal space. In general there is no guarantee that the newly obtained point in the primal space lies in our constraint set $\mathcal{C}$ and, hence, an additional projection may be required. Summarised we get the following algorithm, where $\nabla\Phi$ is the invertible map that connects primal and dual.

Algorithm 1: Projected Mirror Descent

Input: x_1, $\{\eta_s\}_{s \geq 1}$, stopping time t;
for $s = 1, \ldots, t$ **do**
  $\nabla\Phi(\tilde{x}_{s+1}) = \nabla\Phi(x_s) - \eta_s g_s$, where $g_s \in \partial f(x_s)$,
  $x_{s+1} = \Pi_{\mathcal{C}}^{\Phi}(\tilde{x}_{s+1})$.
end

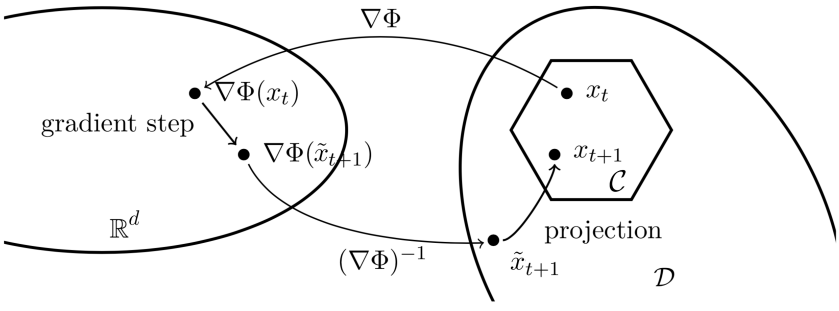

Figure 10.1: Representation of projected mirror descent. From [2].

10.3 Mirror Maps, Bregman Divergence, and Bregman Projection

We now introduce the formal setup needed to property define mirror descent. Note that all points lie in $\mathbb{R}^d$ so the notions of "primal" and "dual" spaces only have an intuitive meaning in the current setting.

Definition 10.5 (Mirror map) *Let $\mathcal{D} \subset \mathbb{R}^d$ be a convex open set such that $\mathcal{C} \subset \overline{\mathcal{D}}$ (where $\overline{\mathcal{D}}$ is the closure of $\mathcal{D}$) and $\mathcal{C} \cap \mathcal{D} \neq \emptyset$. A function $\Phi \colon \mathcal{D} \to \mathbb{R}$ is a* mirror map *if the following properties hold:*

i) Φ is strictly convex and differentiable.

ii) The gradient $\nabla\Phi \colon \mathcal{D} \to \mathbb{R}^d$ is a surjective map.

iii) The gradient diverges on the boundary of $\mathcal{D}$, i.e., $\lim_{x \to \partial\mathcal{D}} \|\nabla\Phi(x)\| = \infty$.

In mirror descent, the mirror map Φ (sometimes also called *potential* function) is used to defined a new geometry. In particular, the gradient of the mirror map Φ is used to map points from the primal to the dual. Precisely, a point $x \in \mathcal{C} \cap \mathcal{D}$ is mapped to $\nabla\Phi(x)$, from which one takes a gradient step to get to $\nabla\Phi(x) - \eta\nabla f(x)$. Property *ii)* tells us that $\nabla\Phi$ takes all possible values in $\mathbb{R}^d$, so this allows us to write the new point as $\nabla\Phi(\tilde{x}) = \nabla\Phi(x) - \eta\nabla f(x)$ for some $\tilde{x} \in \mathcal{D}$. The primal point $\tilde{x}$ may lie outside the set of constraints $\mathcal{C}$, in which case one has to project back onto $\mathcal{C}$. The projection associate to mirror descent is the Bergman projection. This projection is defined based on the notion of Bergman divergence, which serves as a proxy for a notion of "distance".

Definition 10.6 (Bregman divergence) *The Bregman divergence associated with a differentiable function $g : \mathbb{R}^d \to \mathbb{R}$ is defined as*

$$\boxed{D^g(x, y) = g(x) - g(y) - \nabla g(y)^\top (x - y)}$$

The Bregman divergence measures the error of the first order linear approximation of the function g. Precisely, $D^g(x, y)$ is the difference between the value of the function g at x and the value of the first-order Taylor expansion of g around point y evaluated at point x. The Bregman divergence is not symmetric in its arguments (hence, it is not a metric!). If the function g is convex, then $D^g(x, y) \geq 0$ for all x, y.

The Bregman projection is defined in terms of the Bregman divergence of a mirror map.

Definition 10.7 (Bregman projection) *The* Bregman projection *associated to a mirror map Φ is given by*

$$\boxed{\Pi_{\mathcal{C}}^{\Phi}(y) = \operatorname*{argmin}_{x \in \mathcal{C} \cap \mathcal{D}} D^{\Phi}(x, y)}$$

Properties *i)* and *iii)* of the mirror map ensure the existence and the uniqueness of this projection.

10.4 Mirror Maps: Examples

We now describe the specific form that mirror descent takes in two examples of interest. Our first example illustrates the generality of the algorithm and its connection to projected subgradient descent. With our second example we go back to the boosting case illustrated in Section 10.1.2. In this case, along with describing the algorithm, we also establish a few properties of interest that, once used within Theorem 10.11 below, will be instrumental to replace the $\sqrt{d}$ rate in (10.3) of gradient descent with the $\sqrt{\log d}$ rate of mirror descent, as we describe at the end of today's lecture.

10.4.1 Euclidean Balls Leading to Projected Subgradient Descent

The projected subgradient descent algorithm is recovered by taking $\mathcal{D} = \mathbb{R}^d$ and by choosing the mirror map as follows:

$$\Phi(x) = \frac{1}{2}\|x\|_2^2.$$

The associated Bregman divergence is

$$D^\Phi(x,y) = \Phi(x) - \Phi(y) - \nabla\Phi(y)^\top(x-y) = \frac{1}{2}\|x\|_2^2 - \frac{1}{2}\|y\|_2^2 - y^\top x + y^\top y = \frac{1}{2}\|x-y\|_2^2,$$

which implies that the Bregman projection $\Pi_{\mathcal{C}}^\Phi$ coincides with the standard Euclidean projection $\Pi_{\mathcal{C}}$.

10.4.2 Negative Entropy Leading to Exponential Gradient Descent

Let $\mathcal{C} = \Delta_d := \{x \in [0,1]^d : \sum_{i=1}^d x_i = 1\}$. Let us choose as mirror map the *negative entropy* defined as

$$\Phi(x) = \sum_{i=1}^d x_i \log x_i,$$

with $\mathcal{D} = \{x \in \mathbb{R}^d \colon x_i > 0, i = 1,\dots,d\}$. As $\nabla\Phi(x) = 1 + \log(x)$ (where the log function is applied component-wise to the vector x), the mirror descent update reads

$$\log(\tilde{x}_{s+1}) = \log(x_s) - \eta g_s$$

or, equivalently, (in vector notation)

$$\tilde{x}_{s+1} = x_s e^{-\eta g_s}.$$

This is formulation is typically called the *exponential gradient descent algorithm*, a.k.a. *exponential weights*. The Bregman divergence reads, for any $x, y \in \Delta_d$,

$$\begin{aligned} D^\Phi(x,y) &= \Phi(x) - \Phi(y) - \nabla\Phi(y)^\top(x-y) \\ &= \sum_{i=1}^d x_i \log x_i - \sum_{i=1}^d y_i \log y_i - \sum_{i=1}^d (x_i - y_i)\log y_i - \sum_{i=1}^d (x_i - y_i) \\ &= \sum_{i=1}^d x_i \log\left(\frac{x_i}{y_i}\right), \end{aligned}$$

where in the last step we used $\sum_{i=1}^d (x_i - y_i) = 0$ as $x, y \in \Delta_d$. Hence, the Bregman divergence coincides with the *relative entropy* or *Kullback-Leibler divergence*. Moreover, the above calculations can be used to show that Φ is 1-strongly convex with respect to the $\|\cdot\|_1$ norm, something that we will need later on. To prove this, we need Pinsker's inequality.

Proposition 10.8 (Pinsker's inequality) *Let $x, y \in \Delta_d$. Then,*

$$\boxed{\|x-y\|_{\text{tv}} := \frac{1}{2}\sum_{i=1}^d |x_i - y_i| \le \sqrt{\frac{1}{2}\sum_{i=1}^d x_i \log\left(\frac{x_i}{y_i}\right)}}$$

Here, $\|\cdot\|_{\text{tv}}$ denotes the total variation norm.

Proof: See **Problem 4.3** in the Problem Sheets. ■

Using Pinsker's inequality we find

$$D^{\Phi}(x,y) = \Phi(x) - \Phi(y) - \nabla\Phi(y)^{\top}(x-y) = \sum_{i=1}^{d} x_i \log\left(\frac{x_i}{y_i}\right) \geq \frac{1}{2}\left(\sum_{i=1}^{d}|x_i - y_i|\right)^2 = \frac{1}{2}\|x-y\|_1^2,$$

which coincides with the statement that Φ is 1-strongly convex with respect to the $\|\cdot\|_1$ norm. Finally, it can be shown (using the KKT optimality conditions) that the Bregman projection in this case amounts to a normalisation step, namely,

$$\Pi_{\mathcal{C}}^{\Phi}(y) = \operatorname*{argmin}_{x\in\mathcal{C}\cap\mathcal{D}} D^{\Phi}(x,y) = \frac{y_i}{\sum_{i=1}^{d} y_i}.$$

10.5 Useful Properties

The proof of the convergence of mirror descent in the case of Lipschitz functions (Theorem 10.11 below) follows the exact same argument as in the proof of the convergence of subgradient descent (Theorem 9.3), modulo introducing analogous properties for the Bregman divergence and projection. In fact, as we are now going to see, while the Bregman divergence is not a metric, it shares a few key similarities with the squared Euclidean distance.

The following property is analogous to the property $2a^{\top}b = \|a\|_2^2 + \|b\|_2^2 - \|a-b\|_2^2$ used in the proof of Theorem 9.3.

Proposition 10.9 *For any differentiable function $g : \mathbb{R}^d \to \mathbb{R}$ we have*

$$\boxed{(\nabla g(x) - \nabla g(y))^{\top}(x-z) = D^g(x,y) + D^g(z,x) - D^g(z,y)} \tag{10.4}$$

The following property is analogous to the one given in Proposition 9.2 for Euclidean projections.

Proposition 10.10 (Non-expansivity) *Let $x \in \mathcal{C}\cap\mathcal{D}$ and $y \in \mathcal{D}$. Then,*

$$\left(\nabla\Phi\left(\Pi_{\mathcal{C}}^{\Phi}(y)\right) - \nabla\Phi(y)\right)^{\top}\left(\Pi_{\mathcal{C}}^{\Phi}(y) - x\right) \leq 0$$

which implies $D^{\Phi}(x,\Pi_{\mathcal{C}}^{\Phi}(y)) + D^{\Phi}(\Pi_{\mathcal{C}}^{\Phi}(y),y) \leq D^{\Phi}(x,y)$ and, in particular,

$$\boxed{D^{\Phi}(x,\Pi_{\mathcal{C}}^{\Phi}(y)) \leq D^{\Phi}(x,y)}$$

Proof: Note that for any x, y we have

$$\nabla_x D^{\Phi}(x,y) = \nabla_x(\Phi(x) - \Phi(y) - \nabla\Phi(y)^{\top}(x-y)) = \nabla\Phi(x) - \nabla\Phi(y).$$

Choosing $x^{\star} = \Pi_{\mathcal{C}}^{\Phi}(y) \in \arg\min_{x\in\mathcal{C}\cap\mathcal{D}} D^{\Phi}(x,y)$, by the first order optimality conditions for convex problems (Proposition 8.10) we have, for any $z \in \mathcal{C}$,

$$0 \geq \nabla_x D^{\Phi}(x^{\star},y)^{\top}(x^{\star}-z) = \left(\nabla\Phi(x^{\star}) - \nabla\Phi(y)\right)^{\top}(x^{\star}-z),$$

which proves the first inequality. An application of Proposition (10.9) yields

$$\left(\nabla\Phi(x^{\star}) - \nabla\Phi(y)\right)^{\top}(x^{\star}-y) = D^{\Phi}(x^{\star},y) + D^{\Phi}(z,x^{\star}) - D^{\Phi}(z,y),$$

which proves the second inequality. ■

10.6 Lipschitz functions

We are now able to prove the main result of this section.

Theorem 10.11 (Projected Mirror Descent—Lipschitz) *Let f be convex and γ-Lipschitz with respect to a given norm $\|\cdot\|$. Let Φ be a α-strongly convex mirror map on $\mathcal{C} \cap \mathcal{D}$ with respect to the norm $\|\cdot\|$. Assume that $x_1 \in \operatorname{argmin}_{x \in \mathcal{C} \cap \mathcal{D}} \Phi(x)$. Then, projected mirror descent with $\eta = \frac{c}{\gamma}\sqrt{\frac{2\alpha}{t}}$ satisfies*

$$\boxed{f\left(\frac{1}{t}\sum_{s=1}^{t} x_s\right) - f(x^\star) \leq c\gamma\sqrt{\frac{2}{\alpha t}}}$$

where $c^2 = \sup_{x \in \mathcal{C} \cap \mathcal{D}} \Phi(x) - \Phi(x_1)$.

Proof: Let $x \in \mathcal{C} \cap \mathcal{D}$. By the definition of the update step in mirror descent we have

$$g_s = \frac{1}{\eta}(\nabla\Phi(x_s) - \nabla\Phi(\tilde{x}_{s+1})).$$

This yields, along with convexity, Proposition (10.9), and Proposition (10.10),

$$\begin{aligned} f(x_s) - f(x) &\leq g_s^\top (x_s - x) \\ &= \frac{1}{\eta}(\nabla\Phi(x_s) - \nabla\Phi(\tilde{x}_{s+1}))^\top (x_s - x) \\ &= \frac{1}{\eta}(D^\Phi(x_s, \tilde{x}_{s+1}) + D^\Phi(x, x_s) - D^\Phi(x, \tilde{x}_{s+1})) \\ &\leq \frac{1}{\eta}(D^\Phi(x_s, \tilde{x}_{s+1}) + D^\Phi(x, x_s) - D^\Phi(x, x_{s+1})). \end{aligned}$$

By the assumption of strong convexity of Φ with respect to the norm $\|\cdot\|$ we have

$$\Phi(\tilde{x}_{s+1}) \geq \Phi(x_s) + \nabla\Phi(x_s)^\top(\tilde{x}_{s+1} - x_s) + \frac{\alpha}{2}\|\tilde{x}_{s+1} - x_s\|^2,$$

and by the assumption of Lipschitz continuity of f with respect to the norm $\|\cdot\|$ we have

$$\|g_s\|_* \leq \gamma.$$

Using these two inequalities, along with Hölder's inequality, we obtain

$$\begin{aligned} D^\Phi(x_s, \tilde{x}_{s+1}) &= \Phi(x_s) - \Phi(\tilde{x}_{s+1}) - \nabla\Phi(\tilde{x}_{s+1})^\top(x_s - \tilde{x}_{s+1}) \\ &\leq (\nabla\Phi(x_s) - \nabla\Phi(\tilde{x}_{s+1}))^\top(x_s - \tilde{x}_{s+1}) - \frac{\alpha}{2}\|\tilde{x}_{s+1} - x_s\|^2 \\ &= \eta g_s^\top(x_s - \tilde{x}_{s+1}) - \frac{\alpha}{2}\|\tilde{x}_{s+1} - x_s\|^2 \\ &\leq \eta\|g_s\|_*\|x_s - \tilde{x}_{s+1}\| - \frac{\alpha}{2}\|\tilde{x}_{s+1} - x_s\|^2 \\ &\leq \eta\gamma\|x_s - \tilde{x}_{s+1}\| - \frac{\alpha}{2}\|\tilde{x}_{s+1} - x_s\|^2 \\ &\leq \frac{\eta^2\gamma^2}{2\alpha}, \end{aligned}$$

where we used the inequality $az - bz^2 \leq \max_{z\in\mathbb{R}}(az - bz^2) = a^2/4b$ for all $z \in \mathbb{R}$. By convexity, we finally obtain

$$\begin{aligned} f\left(\frac{1}{t}\sum_{s=1}^{t} x_s\right) - f(x^\star) &\leq \frac{1}{t}\sum_{s=1}^{t}(f(x_s) - f(x^\star)) \\ &\leq \frac{1}{\eta t}\sum_{s=1}^{t}(D^\Phi(x^\star, x_s) - D^\Phi(x^\star, x_{s+1})) + \frac{\eta\gamma^2}{2\alpha} \\ &= \frac{1}{\eta t}(D^\Phi(x^\star, x_1) - D^\Phi(x^\star, x_{t+1})) + \frac{\eta\gamma^2}{2\alpha} \\ &\leq \frac{D^\Phi(x^\star, x_1)}{\eta t} + \frac{\eta\gamma^2}{2\alpha}, \end{aligned}$$

where we used that

$$D^\Phi(x, x_{s+1}) = \Phi(x) - \Phi(x_{s+1}) - \nabla\Phi(x_{s+1})^\top(x - x_{s+1}) \geq 0.$$

The proof follows by optimizing the bound over η, and using that

$$D^\Phi(x^\star, x_1) = \Phi(x^\star) - \Phi(x_1) - \nabla\Phi(x_1)^\top(x^\star - x_1) \leq \Phi(x^\star) - \Phi(x_1) \leq \sup_{x\in\mathcal{C}\cap\mathcal{D}} \Phi(x) - \Phi(x_1),$$

where we used the optimality condition in Proposition 8.10 to claim that

$$\nabla\Phi(x_1)^\top(x^\star - x_1) \geq 0$$

as $x_1 \in \operatorname{argmin}_{x\in\mathcal{C}\cap\mathcal{D}} \Phi(x)$ by assumption.

■

10.7 Back to Learning: Linear Predictors with Constraints in Δ_d

We are finally able to show how mirror descent achieves the $\sqrt{\log d}$ rate in the case of the optimization error for boosting, replacing the $\sqrt{d}$ term in the bound (10.3) for gradient descent. Let us consider the setting of Section 10.1.2, and assume that we want to apply mirror descent to solve problem (10.1).

- Let us choose as mirror map the negative entropy $\Phi(w) = \sum_{i=1}^d w_i \log w_i$.
- The starting point of mirror descent reads

  $$w_1 \in \operatorname*{argmin}_{w\in\mathcal{C}\cap\mathcal{D}} \Phi(w) = \frac{1}{d}\mathbf{1},$$

  where $\mathbf{1} \in \mathbb{R}^d$ denotes the all-ones vector. This follows as the minimum of $\sum_{i=1}^d w_i \log w_i$ is achieved at $w_i = 1/d$ for any $i \in [d]$
- As $\Phi(w) \leq 0$ for any $w \in \Delta_d$, we have

  $$c^2 = \sup_{w\in\mathcal{C}\cap\mathcal{D}} \Phi(w) - \Phi(w_1) = \log d.$$

- In Section 10.4.2 we established that Φ is 1-strongly convex with respect to the $\|\cdot\|_1$ norm. So we can choose $\alpha = 1$.

- Hölder's inequality yields, for any $w, u \in \Delta_d$,

$$|R(w) - R(u)| \le \frac{1}{n}\sum_{i=1}^{n} |\varphi(w^\top X_i Y_i) - \varphi(u^\top X_i Y_i)| \le \frac{\gamma_\varphi}{n}\sum_{i=1}^{n} |Y_i (w-u)^\top X_i| \le \gamma_\varphi c_\infty^{\mathcal{X}} \|w-u\|_1,$$

  where we used that $|Y_i| = 1$ and $\|X_i\|_\infty \le c_\infty^{\mathcal{X}}$. Hence, the empirical risk R is γ-Lipschitz with respect to the $\|\cdot\|_1$ norm, with $\gamma = \gamma_\varphi c_\infty^{\mathcal{X}}$.

A direct application of Theorem 10.11 shows that if we apply mirror descent for t time steps with step size

$$\eta = \frac{c}{\gamma}\sqrt{\frac{2\alpha}{t}} = \frac{1}{\gamma_\varphi c_\infty^{\mathcal{X}}}\sqrt{\frac{2\log d}{t}},$$

we obtain (recall $c_1^{\mathcal{W}} = 1$)

$$\boxed{\texttt{Optimization}_\Delta := R(\overline{W}_t) - R(W_\Delta^\star) \le c\gamma\sqrt{\frac{2}{\alpha t}} = c_\infty^{\mathcal{X}} c_1^{\mathcal{W}} \gamma_\varphi \sqrt{\frac{2\log d}{t}}}$$

As in the Euclidean setting, this upper bound precisely matches (modulo universal constants) the upper bound for the `Statistics` term given in (10.2). This suggests to run mirror descent for $t \sim n$ time steps.

References

[1] Olivier Bousquet and Léon Bottou. The tradeoffs of large scale learning. In *Advances in Neural Information Processing Systems*, pages 161–168, 2008.

[2] Sébastien Bubeck et al. Convex optimization: Algorithms and complexity. *Foundations and Trends® in Machine Learning*, 8(3-4):231–357, 2015.