Berkeley
UNIVERSITY OF CALIFORNIA

Optimization

Notes By: Yuhang Cai

2024

CONTENTS

CHAPTER 1

MIRROR DESCENT

Gradient descent algorithm works well for smooth convex optimization problems. However, it's not the best for specific classes of fucntions and bodies: for instance, for minimizing linear functions over the probablity simplex. The general gradient descent algorithm does significantly worse than thte specialized Hedge algorithm.

Another example is about the curse of dimension. In high dimensions, the dimension-free oracle complexity is possible when the objective function f and the constrained set $\mathcal{X}$ are well-behaved in the Euclidean norm. However, if this assumption is not met, then the gradient descent techniques may lose their dimension-free convergence rates. For instance, if we consider a differnetiable convex function f defined on the Euclidean ball s.t. $\|\nabla f(x)\|_\infty \leqslant 1, \forall x \in B_{2,n}$. This implies $\|\nabla f(x)\|_2 \leqslant \sqrt{n}$ and thus the projected gradient descent will converge to the minimum of f on $B_{2,n}$ at a rate of $\sqrt{\frac{n}{t}}$. Known as mirror descent, the rate could be improved to $\sqrt{\frac{\log n}{t}}$.

This suggests asking: *can we somehow change gradient descent to adapt to the "geometry" of the problem?* This is the question that the Mirror Descent algorithm answers.

1.1 Mirror Descent: the Proximal Point View

We know the proximal gradient descent algorithm.

Algorithm 1.1: Proximal Gradient Descent Algorithm

x_1 starting point ;
for $t = 1, 2, \ldots, T$ **do**
  $x_{t+1} = \text{argmin}_x \, \eta\langle \nabla f_t(x_t), x\rangle + \frac{1}{2}\|x - x_t\|^2$;

If we take the derivative, we can get the update rule:

$$\eta \nabla f_t(x_t) + x_{t+1} - x_t = 0 \Longrightarrow x_{t+1} = x_t - \eta \nabla f_t(x_t),$$

which matches the normal gradient descent algorithm. The intuition also makes sense: if we want to minimize the function f_t, we could try to minimize its linear approximiation $f_t(x) + \langle \nabla f_t(x_t), x - x_t\rangle$ instead. But we should be careful not to "over-fit": the linear approximation is good only close to the point x_t. So we can add in a penalty term $\frac{1}{2}\|x - x_t\|^2$ to prevent the linear approximation from being too far off. This means we should minimize:

$$x_{t+1} = \arg\min_x \left\{ f_t(x_t) + \langle \nabla f_t(x_t), x - x_t\rangle + \frac{1}{2}\Big| x - x_t\|^2 \right\}.$$

If we drop the terms that don't depend on x,

$$x_{t+1} = \arg\min_x \left\{ \langle \nabla f_t(x_t), x\rangle + \frac{1}{2}\|x - x_t\|^2 \right\} \tag{1.1}$$

If we have a constrained problem, we can change it to the following form:

$$x_{t+1} = \arg\min_{x \in K} \left\{ \eta \langle \nabla f_t(x_t), x \rangle + \frac{1}{2} \|x - x_t\|^2 \right\} \tag{1.2}$$

Given this persepctive, we can now replace the squared Euclidean norm by other distances to get different algorithms. A particularly useful class of distance functions are Bregman divergences, which we now define and use.

1.1.1 Bregman divergence

Definition 1.1 (Bregman divergence).
Given a strictly convex function h, the Bregman divergence from x to y with respect to function h is:

$$D_h(y\|x) = h(y) - h(x) - \langle \nabla h(x), y - x \rangle.$$

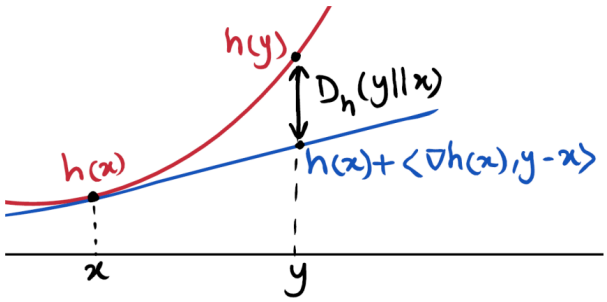

Figure 1.1: $D_h(y\|x)$ for the function $h : \mathbb{R} \to \mathbb{R}$

Example 1.2 (Examples of Bregman Divergence).

- For the function $h(x) = \frac{1}{2}\|x\|^2$, the Bregman divergence is

$$D_h(y\|x) = \frac{1}{2}\|y\|^2 - \frac{1}{2}\|x\|^2 - \langle x, y - x \rangle = \frac{1}{2}\|y - x\|^2.$$

- For the negative entropy function $h(x) = \sum_{i=1}^n (x_i \ln x_i - x_i)$, the Bregman divergence is

$$D_h(y\|x) = \sum_{i=1}^n y_i \ln \frac{y_i}{x_i} - y_i + x_i.$$

  Assuming that $\sum_i y_i = \sum_i x_i = 1$, we can simplify this to $\sum_i y_i \ln \frac{y_i}{x_i}$, which is the KL divergence.

Here are some properties of Bregman divergence.

Lemma 1.3 (Three-point property).
For $x, y, z \in \mathrm{dom}(h)$, we have

$$D_h(x\|y) + D_h(z\|x) - D_h(z\|y) = \langle \nabla h(y) - \nabla h(x), z - x \rangle.$$

Lemma 1.4 (Pythagoras theorem).
Suppose that C is a convex set, $x \in C$ and $y \in \mathbb{R}^d$. Then,

$$D_h\left(x \| \Pi_C(y)\right) + D_h\left(\Pi_C(y) \| y\right) \leqslant D_h(x \| y),$$

where

$$\Pi_C^h(y) = \arg\min_{x \in C} D_h(x \| y).$$

Proof.
In terms of the definition of $\Pi_C(y)$, we have

$$\left(\nabla h\left(\Pi_C(y)\right) - \nabla h(y)\right)^T \left(\Pi_C(y) - x\right) \leqslant 0,$$

for any $x \in C$. □

1.1.2 Changing the distance function

We can replace $\frac{1}{2}\|x-y\|^2$ in (1.1) with a generic Bregman divergence. Let's consider the unconstrained problem, then the update is:

$$x_{t+1} = \arg\min_x \left\{\eta \left\langle \nabla f_t\left(x_t\right), x\right\rangle + D_h\left(x \| x_t\right)\right\}.$$

Taking the derivative, we get

$$\eta \nabla f_t\left(x_t\right) + \nabla h\left(x_{t+1}\right) - \nabla h\left(x_t\right) = 0.$$

This gives us the update rule:

$$x_{t+1} = \nabla h^{-1}\left(\nabla h\left(x_t\right) - \eta \nabla f_t\left(x_t\right)\right). \tag{1.3}$$

Algorithm 1.2: Mirror Descent Algorithm

```
x_1 starting point ;
for t = 1, 2, ..., T do
    x_{t+1} = argmin_{x∈K} {η⟨∇f_t(x_t), x⟩ + D_h(x‖x_t)} ;
```

Example 1.5 (Examples of mirror descent).

- When $h(x) = \frac{1}{2}\|x\|^2$, the gradient $\nabla h(x) = x$, and the update rule becomes

  $$x_{t+1} = x_t - \eta \nabla f_t\left(x_t\right),$$

  the standard gradient descent update.
- When $h(x) = \sum_{i=1}^n \left(x_i \ln x_i - x_i\right)$, then $\nabla h(x) = \left(\ln x_1, \ldots, \ln x_n\right)$, and the update rule is

  $$\left(x_{t+1}\right)_i = \exp\left(\ln\left(x_t\right)_i - \eta \nabla f_t\left(x_t\right)_i\right) = \left(x_t\right)_i e^{-\eta \nabla f_t\left(x_t\right)_i}.$$

  This is exactly the update of the Hedge algorithm.

The same ideas also hold for constrained convex minimization: we now have to search for the minimizer with the set K. In this case, the algorithm using negative entropy resultsin the same Hedge-like update, following by scaling the point down to geta probability vector, thereby giving the probability values in Hedge.

Note 1.6.
What would be the best choice of h to minimize the function f ? It would be $h = \eta f$, because adding $D_f(x\|x_t)$ to the linear approximation of f at x_t gives us back f. Of course, the update now requires us to minize $f(x)$, which is the origional problem. So we should coose an h that is similar to f.

In summary, the algorithm tries to minimize the linear approximation of the function f_t with respect to the Bregman divergence. Depending on the choice of the Bregman divergence, we can get different algorithms - this is the mirror descent framework.

1.2 Mirror Descent: The Mirror Map View

A different view of the mirror descent framework is the one originally presented by Nemirovsk and Yudin. They observe that in gradient descent, at each step we set $x_{t+1} = x_t - \eta f_t(x_t)$. However, **the gradient was actually defined as a linear functional on $\mathbb{R}^n$ and hence naturally belongs to the dual space of $\mathbb{R}^n$**. The fact that we represent this functional as a vecotr is a matter of convenience.

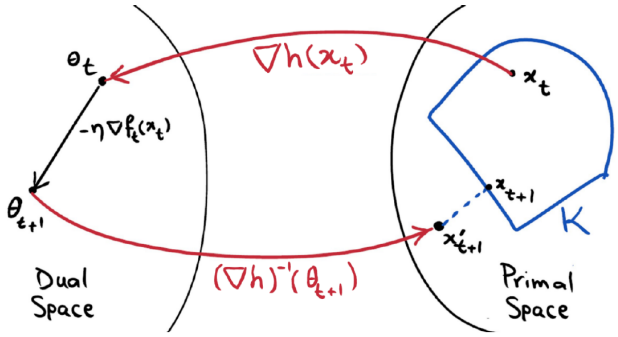

Figure 1.2: The mirror map view of mirror descent

In the vanilla graident descent method, we were working in $\mathbb{R}^n$ endowed with ℓ_2-norm, and this normed space is self-dual. But when working with other normed spaces, adding a covector $\nabla f_t(x_t)$ to a vector x_t might not be the right thing to do. Instead, Nemirovski and Yudin propose the following:

1. We map our current x_t to a point θ_t in the dual space using a mirror map.
2. Next we take the gradient step $\theta_{t+1} = \theta_t - \eta\nabla f_t(x_t)$.
3. We map θ_{t+1} back to a point in the primal sapce x'_{t+1} using the inverse of the mirror map from Step 1 .
4. If we are in the constrained case, this point x'_{t+1} might not be in the convex feasible region K, so we project it to a close point x_{t+1} in K.

The name of the process comes from thinking of the dual space as being a mirror image of the primal space.

1.2.1 Norms and their duals

We skip the definition of norms. Now we consider a Hilbert space.

Definition 1.7 (Dual norm).
Given a linear space C with norm $\|\cdot\|$, the dual norm $\|\cdot\|_* : C^* \to \mathbb{R}$ is defined as:

$$\|y\|_* = \sup\{\langle x, y\rangle \mid \|x\| \leqslant 1, x \in C\}.$$

Example 1.8 (Exampls of dual norms).

- Dual norm of ℓ_2-norm is itself.
- Dual norm of Euclidean norm is itself.
- Dual norm of ℓ_p-norm is ℓ_q-norm, where $\frac{1}{p}+\frac{1}{q}=1$.
- Dual norm of ℓ_1-norm is ℓ_∞-norm.

Theorem 1.9 (Cauchy-Schwarz for general norms).
Given $x \in C, y \in C^*$, we have

$$\langle x, y\rangle \leqslant \|x\|\|y\|_*.$$

Theorem 1.10 (Self dual).
For a finite-dimensional space with norm $\|\cdot\|$, we have

$$(\|\cdot\|_*)_* = \|\cdot\|.$$

1.2.2 Defining the mirror maps

Now we define the mirror maps bsaed on a compact subspace $\mathcal{X}$ of $\mathbb{R}^n$.

Definition 1.11 (Mirror map).
Let $\mathcal{D} \subset \mathbb{R}^n$ be a convex open set such that $\mathcal{X}$ is included in its closure, that is $\mathcal{X} \subset \overline{\mathcal{D}}$, and $\mathcal{X} \cap \mathcal{D} \neq \varnothing$. We say that $h : \mathcal{D} \to \mathbb{R}$ is a mirror map if it safisfies the following properties:

1. h is strictly convex and differentiable.
2. The gradient of h takes all possible values, that is $\nabla h(\mathcal{D}) = \mathbb{R}^n$.
3. The gradient of h diverges on the boundary of $\mathcal{D}$, that is

$$\lim_{x \to \partial\mathcal{D}} \|\nabla h(x)\| = +\infty.$$

In mirror descent the gradient of the mirror map h is used to map points from the "primal" to the "dual" (note that all points lie in $\mathbb{R}^n$ so the notions of primal and dual spaces only have an intuitive meaning). Precisely a point $x \in \mathcal{X} \cap \mathcal{D}$ is mapped to $\nabla h(x)$, from which one takes a gradient step to get to $\nabla h(x) - \eta \nabla f(x)$. Property (ii) then allows us to write the resulting point as $\nabla h(y) = \nabla h(x) - \eta \nabla f(x)$ for some $y \in \mathcal{D}$. The primal point y may lie outside of the set of constraints $\mathcal{X}$, in which case one has to project back onto $\mathcal{X}$. In mirror descent this projection is done via the Bregman divergence associated to h.

Example 1.12 (Examples of mirror maps).

- $h(x) = \frac{1}{2}\|x\|_2^2$ is 1-strongly convex with respect to $\|\cdot\|_2$.
- $h(x) = \sum_{i=1}^n x_i(\log x_i - 1)$ is 1-strongly convex with respect to $\|\cdot\|_1$.

Then we just repeat the Nemirovski-Yudin process.

1.3 Analysis

Now we focus on the mirror descent with constrained convex minimization. Let $y_{t+1} \in \mathcal{D}$ such that

$$\nabla h(y_{t+1}) = \nabla h(x_t) - \eta \nabla f_t(x_t), \tag{1.4}$$

and

$$x_{t+1} \in \Pi_{\mathcal{X}}^h(y_{t+1}) = \operatorname*{argmin}_{x \in \mathcal{X}} D_h(x \| y_{t+1}). \tag{1.5}$$

Theorem 1.13 (Online mirror descent regret bound).
Let $\|\cdot\|$ be a norm on $\mathbb{R}^n$ and h be an α strongly convex function with respect to $\|\cdot\|$. Given $f_1, \ldots, f_T$ be convex differentiable functions, the mirror descent algorithm starting with x_1 and taking constant step size η in every iteraetion produces $x_1, \ldots, x_T$ such that for any $x^* \in \mathbb{R}^n$,

$$\sum_{t=1}^{T} f_t(x_t) \leqslant \sum_{t=1}^{T} f_t(x^*) + \underbrace{\frac{D_h(x^* \| x_1)}{\eta} + \frac{\eta \sum_{t=1}^{T} \|\nabla f_t(x_t)\|_*^2}{2\alpha}}_{\text{regret}}. \tag{1.6}$$

Proof.
For any $x \in \mathcal{X} \cap \mathcal{D}$, we have

$$\begin{aligned}
& f_t(x_t) - f(x) \\
& \leqslant \nabla f_t(x_t)^T (x_t - x) && \text{Convexity of } f \\
& = \frac{1}{\eta} (\nabla h(x_t) - \nabla h(y_{t+1}))^\top (x_t - y_{t+1}) && \text{By (1.4)} \\
& = \frac{1}{\eta} \Big(D_h(x \| x_t) + D_h(x_t \| y_{t+1}) - D_h(x \| y_{t+1}) \Big) && \text{By Lemma 1.3} \\
& \leqslant \frac{1}{\eta} \Big(D_h(x \| x_t) + D_h(x_t \| y_{t+1}) - D_h(x \| x_{t+1}) - D_h(x_{t+1} \| y_{t+1}) \Big) && \text{By Lemma 1.4.}
\end{aligned}$$

The term $D_h(x\|x_t) - D_h(x\|x_{t+1})$ will lead to a telescopic sum, and it remains to bound the other term as follows:

$$\begin{aligned}
& D_h(x_t \| y_{t+1}) - D_h(x_{t+1} \| y_{t+1}) \\
& = h(x_t - h(x_{t+1}) - \nabla h(y_{t+1})^\top (x_t - x_{t+1}) \\
& \leqslant (\nabla h(x_t) - \nabla h(y_{t+1}))^\top (x_t - x_{t+1}) - \frac{\alpha}{2} \|x_t - x_{t+1}\|^2 && \text{By the strong convexity of } h \\
& = \eta \nabla f_t(x_t)^\top (x_t - x_{t+1}) - \frac{\alpha}{2} \|x_t - x_{t+1}\|^2 && \text{By (1.4)} \\
& \leqslant \eta \|\nabla f_t(x_t)\|_* \cdot \|x_t - x_{t+1}\| - \frac{\alpha}{2} \|x_t - x_{t+1}\|^2 && \text{Cauchy-Schwarz} \\
& \leqslant \frac{\eta^2 \|\nabla f_t(x_t)\|_*^2}{2\alpha}. && \text{By completing the square}
\end{aligned}$$

Now we have proved that:

$$\sum_{t=1}^{T} (f_t(x_t) - f(x)) \leqslant \frac{D_h(x\|x_1)}{\eta} + \eta \frac{\|\nabla f_t(x_t)\|_*^2 T}{2\alpha}.$$

This is equivalent to (1.6). □

Theorem 1.14 (Offline mirror descent regret bound).
Let h be a mirror map which is α-strongly convex on $\mathcal{X} \cap D$ w.r.t. $\|\cdot\|$. Let $R^2 = \sup_{x \in \mathcal{X} \cap \mathcal{D}} D_h(x\|x_1)$, and f be convex and L-Lipschitz w.r.t. $\|\cdot\|$. Then the mirror descent with $\eta = \frac{R}{L}\sqrt{\frac{2\alpha}{T}}$ satisfies

$$f\left(\frac{1}{T}\sum_{t=1}^{T} x_t\right) - f(x^*) \leqslant RL\sqrt{\frac{2}{\alpha T}}.$$

Proof sketch.
Apply Theorem 1.13 with $\eta = \frac{R}{L}\sqrt{\frac{2\alpha}{T}}$. □

Note 1.15.
The MD convergence rate $O(1/\sqrt{t})$ is slow. However, this is the best possible rate one can expect when solving nonsmooth large-scale convex problems represented by FO oracles, or any other oracles providing local information.

In fact, we have the following bad news.

Example 1.16 (Lower bound example).
Consider convex minimization problem

$$x_* = \operatorname*{argmin}_{\|x\| \leqslant R} f(x), \tag{P}$$

where $\|\cdot\|$ is either the norm $\|\cdot\|_p$ on $\mathbb{R}^n$ or the nuclear norm on $\mathbb{R}^{n\times n}$. Let $f_* = \min_{\|x\|\leqslant R} f(x)$ and

$$\mathcal{F}(L) = \{f : \mathbb{R}^n \to \mathbb{R} \mid f \text{ is } L\text{-Lipschitz w.r.t. } \|\cdot\|\},$$

and asumme that when solving (P), we have access to an first order oracle. Then for any $t \leqslant n$ and t-step algorithm $\mathcal{B}$ that solves (P) with an FO oracle, we have

$$\sup_{f\in\mathcal{F}(L)} \left[(fx_\beta(f)) - f_*\right] \geqslant 0.01\frac{LR}{\sqrt{t}}.$$

Here $x_\beta(f)$ is the output of $\mathcal{B}$ on the function f.