Convex Optimization: Algorithms and Complexity

Sébastien Bubeck
Theory Group, Microsoft Research
sebubeck@microsoft.com

Contents

Abstract

This monograph presents the main complexity theorems in convex optimization and their corresponding algorithms. Starting from the fundamental theory of black-box optimization, the material progresses towards recent advances in structural optimization and stochastic optimization. Our presentation of black-box optimization, strongly influenced by Nesterov's seminal book and Nemirovski's lecture notes, includes the analysis of cutting plane methods, as well as (accelerated) gradient descent schemes. We also pay special attention to non-Euclidean settings (relevant algorithms include Frank-Wolfe, mirror descent, and dual averaging) and discuss their relevance in machine learning. We provide a gentle introduction to structural optimization with FISTA (to optimize a sum of a smooth and a simple non-smooth term), saddle-point mirror prox (Nemirovski's alternative to Nesterov's smoothing), and a concise description of interior point methods. In stochastic optimization we discuss stochastic gradient descent, mini-batches, random coordinate descent, and sublinear algorithms. We also briefly touch upon convex relaxation of combinatorial problems and the use of randomness to round solutions, as well as random walks based methods.

1

Introduction

The central objects of our study are convex functions and convex sets in $\mathbb{R}^n$.

Definition 1.1 (Convex sets and convex functions)**.** A set $\mathcal{X} \subset \mathbb{R}^n$ is said to be convex if it contains all of its segments, that is

$$\forall (x, y, \gamma) \in \mathcal{X} \times \mathcal{X} \times [0, 1], \ (1 - \gamma)x + \gamma y \in \mathcal{X}.$$

A function $f : \mathcal{X} \to \mathbb{R}$ is said to be convex if it always lies below its chords, that is

$$\forall (x, y, \gamma) \in \mathcal{X} \times \mathcal{X} \times [0, 1], \ f((1 - \gamma)x + \gamma y) \leq (1 - \gamma)f(x) + \gamma f(y).$$

We are interested in algorithms that take as input a convex set $\mathcal{X}$ and a convex function f and output an approximate minimum of f over $\mathcal{X}$. We write compactly the problem of finding the minimum of f over $\mathcal{X}$ as

$$\text{min. } f(x)$$
$$\text{s.t. } x \in \mathcal{X}.$$

In the following we will make more precise how the set of constraints $\mathcal{X}$ and the objective function f are specified to the algorithm. Before that

we proceed to give a few important examples of convex optimization problems in machine learning.

1.1 Some convex optimization problems in machine learning

Many fundamental convex optimization problems in machine learning take the following form:

$$\min_{x \in \mathbb{R}^n} \sum_{i=1}^{m} f_i(x) + \lambda \mathcal{R}(x), \qquad (1.1)$$

where the functions $f_1, \ldots, f_m, \mathcal{R}$ are convex and $\lambda \geq 0$ is a fixed parameter. The interpretation is that $f_i(x)$ represents the cost of using x on the ith element of some data set, and $\mathcal{R}(x)$ is a regularization term which enforces some "simplicity" in x. We discuss now major instances of (1.1). In all cases one has a data set of the form $(w_i, y_i) \in \mathbb{R}^n \times \mathcal{Y}, i = 1, \ldots, m$ and the cost function f_i depends only on the pair (w_i, y_i). We refer to Hastie et al. [2001], Schölkopf and Smola [2002], Shalev-Shwartz and Ben-David [2014] for more details on the origin of these important problems. The mere objective of this section is to expose the reader to a few concrete convex optimization problems which are routinely solved.

In classification one has $\mathcal{Y} = \{-1, 1\}$. Taking $f_i(x) = \max(0, 1 - y_i x^\top w_i)$ (the so-called hinge loss) and $\mathcal{R}(x) = \|x\|_2^2$ one obtains the SVM problem. On the other hand taking $f_i(x) = \log(1 + \exp(-y_i x^\top w_i))$ (the logistic loss) and again $\mathcal{R}(x) = \|x\|_2^2$ one obtains the (regularized) logistic regression problem.

In regression one has $\mathcal{Y} = \mathbb{R}$. Taking $f_i(x) = (x^\top w_i - y_i)^2$ and $\mathcal{R}(x) = 0$ one obtains the vanilla least-squares problem which can be rewritten in vector notation as

$$\min_{x \in \mathbb{R}^n} \|Wx - Y\|_2^2,$$

where $W \in \mathbb{R}^{m \times n}$ is the matrix with $w_i^\top$ on the ith row and $Y = (y_1, \ldots, y_n)^\top$. With $\mathcal{R}(x) = \|x\|_2^2$ one obtains the ridge regression problem, while with $\mathcal{R}(x) = \|x\|_1$ this is the LASSO problem Tibshirani [1996].

Our last two examples are of a slightly different flavor. In particular the design variable x is now best viewed as a matrix, and thus we

denote it by a capital letter X. The sparse inverse covariance estimation problem can be written as follows, given some empirical covariance matrix Y,

$$\text{min. } \text{Tr}(XY) - \text{logdet}(X) + \lambda\|X\|_1$$
$$\text{s.t. } X \in \mathbb{R}^{n\times n}, X^\top = X, X \succeq 0.$$

Intuitively the above problem is simply a regularized maximum likelihood estimator (under a Gaussian assumption).

Finally we introduce the convex version of the matrix completion problem. Here our data set consists of observations of some of the entries of an unknown matrix Y, and we want to "complete" the unobserved entries of Y in such a way that the resulting matrix is "simple" (in the sense that it has low rank). After some massaging (see Candès and Recht [2009]) the (convex) matrix completion problem can be formulated as follows:

$$\text{min. } \text{Tr}(X)$$
$$\text{s.t. } X \in \mathbb{R}^{n\times n}, X^\top = X, X \succeq 0, X_{i,j} = Y_{i,j} \text{ for } (i,j) \in \Omega,$$

where $\Omega \subset [n]^2$ and $(Y_{i,j})_{(i,j)\in\Omega}$ are given.

1.2   Basic properties of convexity

A basic result about convex sets that we shall use extensively is the Separation Theorem.

Theorem 1.1 (Separation Theorem). Let $\mathcal{X} \subset \mathbb{R}^n$ be a closed convex set, and $x_0 \in \mathbb{R}^n \setminus \mathcal{X}$. Then, there exists $w \in \mathbb{R}^n$ and $t \in \mathbb{R}$ such that

$$w^\top x_0 < t, \text{ and } \forall x \in \mathcal{X}, w^\top x \geq t.$$

Note that if $\mathcal{X}$ is not closed then one can only guarantee that $w^\top x_0 \leq w^\top x, \forall x \in \mathcal{X}$ (and $w \neq 0$). This immediately implies the Supporting Hyperplane Theorem ($\partial\mathcal{X}$ denotes the boundary of $\mathcal{X}$, that is the closure without the interior):

Theorem 1.2 (Supporting Hyperplane Theorem). Let $\mathcal{X} \subset \mathbb{R}^n$ be a convex set, and $x_0 \in \partial\mathcal{X}$. Then, there exists $w \in \mathbb{R}^n, w \neq 0$ such that

$$\forall x \in \mathcal{X}, w^\top x \geq w^\top x_0.$$

We introduce now the key notion of *subgradients*.

Definition 1.2 (Subgradients). Let $\mathcal{X} \subset \mathbb{R}^n$, and $f : \mathcal{X} \to \mathbb{R}$. Then $g \in \mathbb{R}^n$ is a subgradient of f at $x \in \mathcal{X}$ if for any $y \in \mathcal{X}$ one has

$$f(x) - f(y) \leq g^\top (x - y).$$

The set of subgradients of f at x is denoted $\partial f(x)$.

To put it differently, for any $x \in \mathcal{X}$ and $g \in \partial f(x)$, f is above the linear function $y \mapsto f(x) + g^\top (y - x)$. The next result shows (essentially) that a convex functions always admit subgradients.

Proposition 1.1 (Existence of subgradients). Let $\mathcal{X} \subset \mathbb{R}^n$ be convex, and $f : \mathcal{X} \to \mathbb{R}$. If $\forall x \in \mathcal{X}, \partial f(x) \neq \emptyset$ then f is convex. Conversely if f is convex then for any $x \in \mathrm{int}(\mathcal{X}), \partial f(x) \neq \emptyset$. Furthermore if f is convex and differentiable at x then $\nabla f(x) \in \partial f(x)$.

Before going to the proof we recall the definition of the epigraph of a function $f : \mathcal{X} \to \mathbb{R}$:

$$\mathrm{epi}(f) = \{(x, t) \in \mathcal{X} \times \mathbb{R} : t \geq f(x)\}.$$

It is obvious that a function is convex if and only if its epigraph is a convex set.

Proof. The first claim is almost trivial: let $g \in \partial f((1 - \gamma)x + \gamma y)$, then by definition one has

$$f((1 - \gamma)x + \gamma y) \leq f(x) + \gamma g^\top (y - x),$$
$$f((1 - \gamma)x + \gamma y) \leq f(y) + (1 - \gamma)g^\top (x - y),$$

which clearly shows that f is convex by adding the two (appropriately rescaled) inequalities.

Now let us prove that a convex function f has subgradients in the interior of $\mathcal{X}$. We build a subgradient by using a supporting hyperplane to the epigraph of the function. Let $x \in \mathcal{X}$. Then clearly $(x, f(x)) \in \partial \mathrm{epi}(f)$, and $\mathrm{epi}(f)$ is a convex set. Thus by using the Supporting Hyperplane Theorem, there exists $(a, b) \in \mathbb{R}^n \times \mathbb{R}$ such that

$$a^\top x + b f(x) \geq a^\top y + bt, \forall (y, t) \in \mathrm{epi}(f). \tag{1.2}$$

Clearly, by letting t tend to infinity, one can see that $b \leq 0$. Now let us assume that x is in the interior of $\mathcal{X}$. Then for $\varepsilon > 0$ small enough, $y = x + \varepsilon a \in \mathcal{X}$, which implies that b cannot be equal to 0 (recall that if $b = 0$ then necessarily $a \neq 0$ which allows to conclude by contradiction). Thus rewriting (1.2) for $t = f(y)$ one obtains

$$f(x) - f(y) \leq \frac{1}{|b|} a^\top (x - y).$$

Thus $a/|b| \in \partial f(x)$ which concludes the proof of the second claim.

Finally let f be a convex and differentiable function. Then by definition:

$$
\begin{aligned}
f(y) \quad &\geq \quad \frac{f((1 - \gamma)x + \gamma y) - (1 - \gamma)f(x)}{\gamma} \\
&= \quad f(x) + \frac{f(x + \gamma(y - x)) - f(x)}{\gamma} \\
&\underset{\gamma \to 0}{\to} \quad f(x) + \nabla f(x)^\top (y - x),
\end{aligned}
$$

which shows that $\nabla f(x) \in \partial f(x)$. $\qquad\square$

In several cases of interest the set of contraints can have an empty interior, in which case the above proposition does not yield any information. However it is easy to replace int$(\mathcal{X})$ by ri$(\mathcal{X})$ -the relative interior of $\mathcal{X}$- which is defined as the interior of $\mathcal{X}$ when we view it as subset of the affine subspace it generates. Other notions of convex analysis will prove to be useful in some parts of this text. In particular the notion of *closed convex functions* is convenient to exclude pathological cases: these are the convex functions with closed epigraphs. Sometimes it is also useful to consider the extension of a convex function $f : \mathcal{X} \to \mathbb{R}$ to a function from $\mathbb{R}^n$ to $\overline{\mathbb{R}}$ by setting $f(x) = +\infty$ for $x \notin \mathcal{X}$. In convex analysis one uses the term *proper convex function* to denote a convex function with values in $\mathbb{R} \cup \{+\infty\}$ such that there exists $x \in \mathbb{R}^n$ with $f(x) < +\infty$. **From now on all convex functions will be closed, and if necessary we consider also their proper extension.** We refer the reader to Rockafellar [1970] for an extensive discussion of these notions.

1.3 Why convexity?

The key to the algorithmic success in minimizing convex functions is that these functions exhibit a *local to global* phenomenon. We have already seen one instance of this in Proposition 1.1, where we showed that $\nabla f(x) \in \partial f(x)$: the gradient $\nabla f(x)$ contains a priori only local information about the function f around x while the subdifferential $\partial f(x)$ gives a global information in the form of a linear lower bound on the entire function. Another instance of this local to global phenomenon is that local minima of convex functions are in fact global minima:

Proposition 1.2 (Local minima are global minima)**.** Let f be convex. If x is a local minimum of f then x is a global minimum of f. Furthermore this happens if and only if $0 \in \partial f(x)$.

Proof. Clearly $0 \in \partial f(x)$ if and only if x is a global minimum of f. Now assume that x is local minimum of f. Then for γ small enough one has for any y,

$$f(x) \le f((1 - \gamma)x + \gamma y) \le (1 - \gamma)f(x) + \gamma f(y),$$

which implies $f(x) \le f(y)$ and thus x is a global minimum of f. $\square$

The nice behavior of convex functions will allow for very fast algorithms to optimize them. This alone would not be sufficient to justify the importance of this class of functions (after all constant functions are pretty easy to optimize). However it turns out that surprisingly many optimization problems admit a convex (re)formulation. The excellent book Boyd and Vandenberghe [2004] describes in great details the various methods that one can employ to uncover the convex aspects of an optimization problem. We will not repeat these arguments here, but we have already seen that many famous machine learning problems (SVM, ridge regression, logistic regression, LASSO, sparse covariance estimation, and matrix completion) are formulated as convex problems.

We conclude this section with a simple extension of the optimality condition "$0 \in \partial f(x)$" to the case of constrained optimization. We state this result in the case of a differentiable function for sake of simplicity.

Proposition 1.3 (First order optimality condition). Let f be convex and $\mathcal{X}$ a closed convex set on which f is differentiable. Then

$$x^* \in \operatorname*{argmin}_{x \in \mathcal{X}} f(x),$$

if and only if one has

$$\nabla f(x^*)^\top (x^* - y) \le 0, \forall y \in \mathcal{X}.$$

Proof. The "if" direction is trivial by using that a gradient is also a subgradient. For the "only if" direction it suffices to note that if $\nabla f(x)^\top (y - x) < 0$, then f is locally decreasing around x on the line to y (simply consider $h(t) = f(x + t(y - x))$ and note that $h'(0) = \nabla f(x)^\top (y - x)$). $\qquad\square$

1.4   Black-box model

We now describe our first model of "input" for the objective function and the set of constraints. In the black-box model we assume that we have unlimited computational resources, the set of constraint $\mathcal{X}$ is known, and the objective function $f : \mathcal{X} \to \mathbb{R}$ is unknown but can be accessed through queries to *oracles*:

- A zeroth order oracle takes as input a point $x \in \mathcal{X}$ and outputs the value of f at x.

- A first order oracle takes as input a point $x \in \mathcal{X}$ and outputs a subgradient of f at x.

In this context we are interested in understanding the *oracle complexity* of convex optimization, that is how many queries to the oracles are necessary and sufficient to find an ε-approximate minima of a convex function. To show an upper bound on the sample complexity we need to propose an algorithm, while lower bounds are obtained by information theoretic reasoning (we need to argue that if the number of queries is "too small" then we don't have enough information about the function to identify an ε-approximate solution).

From a mathematical point of view, the strength of the black-box model is that it will allow us to derive a *complete* theory of convex optimization, in the sense that we will obtain matching upper and lower bounds on the oracle complexity for various subclasses of interesting convex functions. While the model by itself does not limit our computational resources (for instance any operation on the constraint set $\mathcal{X}$ is allowed) we will of course pay special attention to the algorithms' *computational complexity* (i.e., the number of elementary operations that the algorithm needs to do). We will also be interested in the situation where the set of constraint $\mathcal{X}$ is unknown and can only be accessed through a *separation oracle*: given $x \in \mathbb{R}^n$, it outputs either that x is in $\mathcal{X}$, or if $x \notin \mathcal{X}$ then it outputs a separating hyperplane between x and $\mathcal{X}$.

The black-box model was essentially developed in the early days of convex optimization (in the Seventies) with Nemirovski and Yudin [1983] being still an important reference for this theory (see also Nemirovski [1995]). In the recent years this model and the corresponding algorithms have regained a lot of popularity, essentially for two reasons:

- It is possible to develop algorithms with dimension-free oracle complexity which is quite attractive for optimization problems in very high dimension.

- Many algorithms developed in this model are robust to noise in the output of the oracles. This is especially interesting for stochastic optimization, and very relevant to machine learning applications. We will explore this in details in Chapter 6.

Chapter 2, Chapter 3 and Chapter 4 are dedicated to the study of the black-box model (noisy oracles are discussed in Chapter 6). We do not cover the setting where only a zeroth order oracle is available, also called derivative free optimization, and we refer to Conn et al. [2009], Audibert et al. [2011] for further references on this.

1.5 Structured optimization

The black-box model described in the previous section seems extremely wasteful for the applications we discussed in Section 1.1. Consider for instance the LASSO objective: $x \mapsto \|Wx - y\|_2^2 + \|x\|_1$. We know this function *globally*, and assuming that we can only make local queries through oracles seem like an artificial constraint for the design of algorithms. Structured optimization tries to address this observation. Ultimately one would like to take into account the global structure of both f and $\mathcal{X}$ in order to propose the most efficient optimization procedure. An extremely powerful hammer for this task are the Interior Point Methods. We will describe this technique in Chapter 5 alongside with other more recent techniques such as FISTA or Mirror Prox.

We briefly describe now two classes of optimization problems for which we will be able to exploit the structure very efficiently, these are the LPs (Linear Programs) and SDPs (Semi-Definite Programs). Ben-Tal and Nemirovski [2001] describe a more general class of Conic Programs but we will not go in that direction here.

The class LP consists of problems where $f(x) = c^\top x$ for some $c \in \mathbb{R}^n$, and $\mathcal{X} = \{x \in \mathbb{R}^n : Ax \le b\}$ for some $A \in \mathbb{R}^{m \times n}$ and $b \in \mathbb{R}^m$.

The class SDP consists of problems where the optimization variable is a symmetric matrix $X \in \mathbb{R}^{n \times n}$. Let $\mathbb{S}^n$ be the space of $n \times n$ symmetric matrices (respectively $\mathbb{S}_+^n$ is the space of positive semi-definite matrices), and let $\langle \cdot, \cdot \rangle$ be the Frobenius inner product (recall that it can be written as $\langle A, B \rangle = \mathrm{Tr}(A^\top B)$). In the class SDP the problems are of the following form: $f(x) = \langle X, C \rangle$ for some $C \in \mathbb{R}^{n \times n}$, and $\mathcal{X} = \{X \in \mathbb{S}_+^n : \langle X, A_i \rangle \le b_i, i \in \{1, \ldots, m\}\}$ for some $A_1, \ldots, A_m \in \mathbb{R}^{n \times n}$ and $b \in \mathbb{R}^m$. Note that the matrix completion problem described in Section 1.1 is an example of an SDP.

1.6 Overview of the results and disclaimer

The overarching aim of this monograph is to present the main complexity theorems in convex optimization and the corresponding algorithms. We focus on five major results in convex optimization which give the overall structure of the text: the existence of efficient cutting-plane

methods with optimal oracle complexity (Chapter 2), a complete characterization of the relation between first order oracle complexity and curvature in the objective function (Chapter 3), first order methods beyond Euclidean spaces (Chapter 4), non-black box methods (such as interior point methods) can give a quadratic improvement in the number of iterations with respect to optimal black-box methods (Chapter 5), and finally noise robustness of first order methods (Chapter 6). Table 1.1 can be used as a quick reference to the results proved in Chapter 2 to Chapter 5, as well as some of the results of Chapter 6 (this last chapter is the most relevant to machine learning but the results are also slightly more specific which make them harder to summarize).

An important disclaimer is that the above selection leaves out methods derived from duality arguments, as well as the two most popular research avenues in convex optimization: (i) using convex optimization in non-convex settings, and (ii) practical large-scale algorithms. Entire books have been written on these topics, and new books have yet to be written on the impressive collection of new results obtained for both (i) and (ii) in the past five years.

A few of the blatant omissions regarding (i) include (a) the theory of submodular optimization (see Bach [2013]), (b) convex relaxations of combinatorial problems (a short example is given in Section 6.6), and (c) methods inspired from convex optimization for non-convex problems such as low-rank matrix factorization (see e.g. Jain et al. [2013] and references therein), neural networks optimization, etc.

With respect to (ii) the most glaring omissions include (a) heuristics (the only heuristic briefly discussed here is the non-linear conjugate gradient in Section 2.4), (b) methods for distributed systems, and (c) adaptivity to unknown parameters. Regarding (a) we refer to Nocedal and Wright [2006] where the most practical algorithms are discussed in great details (e.g., quasi-newton methods such as BFGS and L-BFGS, primal-dual interior point methods, etc.). The recent survey Boyd et al. [2011] discusses the alternating direction method of multipliers (ADMM) which is a popular method to address (b). Finally (c) is a subtle and important issue. In the entire monograph the emphasis is on presenting the algorithms and proofs in the simplest way, and

thus for sake of convenience we assume that the relevant parameters describing the regularity and curvature of the objective function (Lipschitz constant, smoothness constant, strong convexity parameter) are known and can be used to tune the algorithm's own parameters. Line search is a powerful technique to replace the knowledge of these parameters and it is heavily used in practice, see again Nocedal and Wright [2006]. We observe however that from a theoretical point of view (c) is only a matter of logarithmic factors as one can always run in parallel several copies of the algorithm with different guesses for the values of the parameters[1]. Overall the attitude of this text with respect to (ii) is best summarized by a quote of Thomas Cover: "theory is the first term in the Taylor series of practice", Cover [1992].

Notation. We always denote by x^* a point in $\mathcal{X}$ such that $f(x^*) = \min_{x \in \mathcal{X}} f(x)$ (note that the optimization problem under consideration will always be clear from the context). In particular we always assume that x^* exists. For a vector $x \in \mathbb{R}^n$ we denote by $x(i)$ its ith coordinate. The dual of a norm $\|\cdot\|$ (defined later) will be denoted either $\|\cdot\|_*$ or $\|\cdot\|^*$ (depending on whether the norm already comes with a subscript). Other notation are standard (e.g., I_n for the $n \times n$ identity matrix, $\succeq$ for the positive semi-definite order on matrices, etc).

---

[1]Note that this trick does not work in the context of Chapter 6.

f	Algorithm	Rate	# Iter	Cost/iter
non-smooth	center of gravity	$\exp\left(-\frac{t}{n}\right)$	$n\log\left(\frac{1}{\varepsilon}\right)$	1 ∇, 1 n-dim $\int$
non-smooth	ellipsoid method	$\frac{R}{r}\exp\left(-\frac{t}{n^2}\right)$	$n^2\log\left(\frac{R}{r\varepsilon}\right)$	1 ∇, mat-vec $\times$
non-smooth	Vaidya	$\frac{Rn}{r}\exp\left(-\frac{t}{n}\right)$	$n\log\left(\frac{Rn}{r\varepsilon}\right)$	1 ∇, mat-mat $\times$
quadratic	CG	exact $\exp\left(-\frac{t}{\kappa}\right)$	n $\kappa\log\left(\frac{1}{\varepsilon}\right)$	1 ∇
non-smooth, Lipschitz	PGD	$RL/\sqrt{t}$	R^2L^2/ε^2	1 ∇, 1 proj.
smooth	PGD	$\beta R^2/t$	$\beta R^2/\varepsilon$	1 ∇, 1 proj.
smooth	AGD	$\beta R^2/t^2$	$R\sqrt{\beta/\varepsilon}$	1 ∇
smooth (any norm)	FW	$\beta R^2/t$	$\beta R^2/\varepsilon$	1 ∇, 1 LP
strong. conv., Lipschitz	PGD	$L^2/(\alpha t)$	$L^2/(\alpha\varepsilon)$	1 ∇ , 1 proj.
strong. conv., smooth	PGD	$R^2\exp\left(-\frac{t}{\kappa}\right)$	$\kappa\log\left(\frac{R^2}{\varepsilon}\right)$	1 ∇ , 1 proj.
strong. conv., smooth	AGD	$R^2\exp\left(-\frac{t}{\sqrt{\kappa}}\right)$	$\sqrt{\kappa}\log\left(\frac{R^2}{\varepsilon}\right)$	1 ∇
$f+g$, f smooth, g simple	FISTA	$\beta R^2/t^2$	$R\sqrt{\beta/\varepsilon}$	1 ∇ of f Prox of g
$\max_{y\in\mathcal{Y}}\varphi(x,y)$, φ smooth	SP-MP	$\beta R^2/t$	$\beta R^2/\varepsilon$	MD on $\mathcal{X}$ MD on $\mathcal{Y}$
linear, $\mathcal{X}$ with F ν-self-conc.	IPM	$\nu\exp\left(-\frac{t}{\sqrt{\nu}}\right)$	$\sqrt{\nu}\log\left(\frac{\nu}{\varepsilon}\right)$	Newton step on F
non-smooth	SGD	$BL/\sqrt{t}$	B^2L^2/ε^2	1 stoch. ∇, 1 proj.
non-smooth, strong. conv.	SGD	$B^2/(\alpha t)$	$B^2/(\alpha\varepsilon)$	1 stoch. ∇, 1 proj.
$f=\frac{1}{m}\sum f_i$ f_i smooth strong. conv.	SVRG	$-$	$(m+\kappa)\log\left(\frac{1}{\varepsilon}\right)$	1 stoch. ∇

Table 1.1: Summary of the results proved in Chapter 2 to Chapter 5 and some of the results in Chapter 6.

2

Convex optimization in finite dimension

Let $\mathcal{X} \subset \mathbb{R}^n$ be a convex body (that is a compact convex set with non-empty interior), and $f : \mathcal{X} \to [-B, B]$ be a continuous and convex function. Let $r, R > 0$ be such that $\mathcal{X}$ is contained in an Euclidean ball of radius R (respectively it contains an Euclidean ball of radius r). In this chapter we give several black-box algorithms to solve

$$\begin{aligned} &\min. \; f(x) \\ &\text{s.t. } x \in \mathcal{X}. \end{aligned}$$

As we will see these algorithms have an oracle complexity which is linear (or quadratic) in the dimension, hence the title of the chapter (in the next chapter the oracle complexity will be *independent* of the dimension). An interesting feature of the methods discussed here is that they only need a separation oracle for the constraint set $\mathcal{X}$. In the literature such algorithms are often referred to as *cutting plane methods*. In particular these methods can be used to *find* a point $x \in \mathcal{X}$ given only a separating oracle for $\mathcal{X}$ (this is also known as the *feasibility problem*).

2.1 The center of gravity method

We consider the following simple iterative algorithm[1]: let $\mathcal{S}_1 = \mathcal{X}$, and for $t \geq 1$ do the following:

1. Compute

$$c_t = \frac{1}{\text{vol}(\mathcal{S}_t)} \int_{x \in \mathcal{S}_t} x dx. \tag{2.1}$$

2. Query the first order oracle at c_t and obtain $w_t \in \partial f(c_t)$. Let

$$\mathcal{S}_{t+1} = \mathcal{S}_t \cap \{x \in \mathbb{R}^n : (x - c_t)^\top w_t \leq 0\}.$$

If stopped after t queries to the first order oracle then we use t queries to a zeroth order oracle to output

$$x_t \in \operatorname*{argmin}_{1 \leq r \leq t} f(c_r).$$

This procedure is known as the *center of gravity method*, it was discovered independently on both sides of the Wall by Levin [1965] and Newman [1965].

Theorem 2.1. The center of gravity method satisfies

$$f(x_t) - \min_{x \in \mathcal{X}} f(x) \leq 2B \left(1 - \frac{1}{e}\right)^{t/n}.$$

Before proving this result a few comments are in order.

To attain an ε-optimal point the center of gravity method requires $O(n \log(2B/\varepsilon))$ queries to both the first and zeroth order oracles. It can be shown that this is the best one can hope for, in the sense that for ε small enough one needs $\Omega(n \log(1/\varepsilon))$ calls to the oracle in order to find an ε-optimal point, see Nemirovski and Yudin [1983] for a formal proof.

The rate of convergence given by Theorem 2.1 is exponentially fast. In the optimization literature this is called a *linear rate* as the (estimated) error at iteration $t+1$ is linearly related to the error at iteration t.

---

[1]As a warm-up we assume in this section that $\mathcal{X}$ is known. It should be clear from the arguments in the next section that in fact the same algorithm would work if initialized with $\mathcal{S}_1 \supset \mathcal{X}$.

The last and most important comment concerns the computational complexity of the method. It turns out that finding the center of gravity c_t is a very difficult problem by itself, and we do not have computationally efficient procedure to carry out this computation in general. In Section 6.7 we will discuss a relatively recent (compared to the 50 years old center of gravity method!) randomized algorithm to approximately compute the center of gravity. This will in turn give a randomized center of gravity method which we will describe in detail.

We now turn to the proof of Theorem 2.1. We will use the following elementary result from convex geometry:

Lemma 2.2 (Grünbaum [1960])**.** Let $\mathcal{K}$ be a centered convex set, i.e., $\int_{x \in \mathcal{K}} x dx = 0$, then for any $w \in \mathbb{R}^n, w \neq 0$, one has

$$\text{Vol}\left(\mathcal{K} \cap \{x \in \mathbb{R}^n : x^\top w \geq 0\}\right) \geq \frac{1}{e}\text{Vol}(\mathcal{K}).$$

We now prove Theorem 2.1.

Proof. Let x^* be such that $f(x^*) = \min_{x \in \mathcal{X}} f(x)$. Since $w_t \in \partial f(c_t)$ one has

$$f(c_t) - f(x) \leq w_t^\top (c_t - x).$$

and thus

$$\mathcal{S}_t \setminus \mathcal{S}_{t+1} \subset \{x \in \mathcal{X} : (x - c_t)^\top w_t > 0\} \subset \{x \in \mathcal{X} : f(x) > f(c_t)\}, \quad (2.2)$$

which clearly implies that one can never remove the optimal point from our sets in consideration, that is $x^* \in \mathcal{S}_t$ for any t. Without loss of generality we can assume that we always have $w_t \neq 0$, for otherwise one would have $f(c_t) = f(x^*)$ which immediately conludes the proof. Now using that $w_t \neq 0$ for any t and Lemma 2.2 one clearly obtains

$$\text{vol}(\mathcal{S}_{t+1}) \leq \left(1 - \frac{1}{e}\right)^t \text{vol}(\mathcal{X}).$$

For $\varepsilon \in [0, 1]$, let $\mathcal{X}_\varepsilon = \{(1 - \varepsilon)x^* + \varepsilon x, x \in \mathcal{X}\}$. Note that $\text{vol}(\mathcal{X}_\varepsilon) = \varepsilon^n \text{vol}(\mathcal{X})$. These volume computations show that for $\varepsilon > \left(1 - \frac{1}{e}\right)^{t/n}$ one has $\text{vol}(\mathcal{X}_\varepsilon) > \text{vol}(\mathcal{S}_{t+1})$. In particular this implies that for $\varepsilon > \left(1 - \frac{1}{e}\right)^{t/n}$, there must exist a time $r \in \{1, \ldots, t\}$, and $x_\varepsilon \in \mathcal{X}_\varepsilon$, such

that $x_\varepsilon \in \mathcal{S}_r$ and $x_\varepsilon \notin \mathcal{S}_{r+1}$. In particular by (2.2) one has $f(c_r) < f(x_\varepsilon)$. On the other hand by convexity of f one clearly has $f(x_\varepsilon) \leq f(x^*) + 2\varepsilon B$. This concludes the proof. $\qquad\square$

2.2 The ellipsoid method

Recall that an ellipsoid is a convex set of the form

$$\mathcal{E} = \{x \in \mathbb{R}^n : (x - c)^\top H^{-1}(x - c) \leq 1\},$$

where $c \in \mathbb{R}^n$, and H is a symmetric positive definite matrix. Geometrically c is the center of the ellipsoid, and the semi-axes of $\mathcal{E}$ are given by the eigenvectors of H, with lengths given by the square root of the corresponding eigenvalues.

We give now a simple geometric lemma, which is at the heart of the ellipsoid method.

Lemma 2.3. Let $\mathcal{E}_0 = \{x \in \mathbb{R}^n : (x - c_0)^\top H_0^{-1}(x - c_0) \leq 1\}$. For any $w \in \mathbb{R}^n$, $w \neq 0$, there exists an ellipsoid $\mathcal{E}$ such that

$$\mathcal{E} \supset \{x \in \mathcal{E}_0 : w^\top(x - c_0) \leq 0\}, \tag{2.3}$$

and

$$\mathrm{vol}(\mathcal{E}) \leq \exp\left(-\frac{1}{2n}\right)\mathrm{vol}(\mathcal{E}_0). \tag{2.4}$$

Furthermore for $n \geq 2$ one can take $\mathcal{E} = \{x \in \mathbb{R}^n : (x-c)^\top H^{-1}(x-c) \leq 1\}$ where

$$c = c_0 - \frac{1}{n+1}\frac{H_0 w}{\sqrt{w^\top H_0 w}}, \tag{2.5}$$

$$H = \frac{n^2}{n^2 - 1}\left(H_0 - \frac{2}{n+1}\frac{H_0 w w^\top H_0}{w^\top H_0 w}\right). \tag{2.6}$$

Proof. For $n = 1$ the result is obvious, in fact we even have $\mathrm{vol}(\mathcal{E}) \leq \frac{1}{2}\mathrm{vol}(\mathcal{E}_0)$.

For $n \geq 2$ one can simply verify that the ellipsoid given by (2.5) and (2.6) satisfy the required properties (2.3) and (2.4). Rather than bluntly doing these computations we will show how to derive (2.5) and (2.6). As a by-product this will also show that the ellipsoid defined by

(2.5) and (2.6) is the unique ellipsoid of minimal volume that satisfy (2.3). Let us first focus on the case where $\mathcal{E}_0$ is the Euclidean ball $\mathcal{B} = \{x \in \mathbb{R}^n : x^\top x \leq 1\}$. We momentarily assume that w is a unit norm vector.

By doing a quick picture, one can see that it makes sense to look for an ellipsoid $\mathcal{E}$ that would be centered at $c = -tw$, with $t \in [0, 1]$ (presumably t will be small), and such that one principal direction is w (with inverse squared semi-axis $a > 0$), and the other principal directions are all orthogonal to w (with the same inverse squared semi-axes $b > 0$). In other words we are looking for $\mathcal{E} = \{x : (x-c)^\top H^{-1}(x-c) \leq 1\}$ with

$$c = -tw, \text{ and } H^{-1} = aww^\top + b(\mathrm{I}_n - ww^\top).$$

Now we have to express our constraints on the fact that $\mathcal{E}$ should contain the half Euclidean ball $\{x \in \mathcal{B} : x^\top w \leq 0\}$. Since we are also looking for $\mathcal{E}$ to be as small as possible, it makes sense to ask for $\mathcal{E}$ to "touch" the Euclidean ball, both at $x = -w$, and at the equator $\partial \mathcal{B} \cap w^\perp$. The former condition can be written as:

$$(-w - c)^\top H^{-1}(-w - c) = 1 \Leftrightarrow (t - 1)^2 a = 1,$$

while the latter is expressed as:

$$\forall y \in \partial \mathcal{B} \cap w^\perp, (y - c)^\top H^{-1}(y - c) = 1 \Leftrightarrow b + t^2 a = 1.$$

As one can see from the above two equations, we are still free to choose any value for $t \in [0, 1/2)$ (the fact that we need $t < 1/2$ comes from $b = 1 - \left(\frac{t}{t-1}\right)^2 > 0$). Quite naturally we take the value that minimizes the volume of the resulting ellipsoid. Note that

$$\frac{\mathrm{vol}(\mathcal{E})}{\mathrm{vol}(\mathcal{B})} = \frac{1}{\sqrt{a}}\left(\frac{1}{\sqrt{b}}\right)^{n-1} = \frac{1}{\sqrt{\frac{1}{(1-t)^2}\left(1 - \left(\frac{t}{1-t}\right)^2\right)^{n-1}}} = \frac{1}{\sqrt{f\left(\frac{1}{1-t}\right)}},$$

where $f(h) = h^2(2h - h^2)^{n-1}$. Elementary computations show that the maximum of f (on $[1, 2]$) is attained at $h = 1 + \frac{1}{n}$ (which corresponds to $t = \frac{1}{n+1}$), and the value is

$$\left(1 + \frac{1}{n}\right)^2\left(1 - \frac{1}{n^2}\right)^{n-1} \geq \exp\left(\frac{1}{n}\right),$$

where the lower bound follows again from elementary computations. Thus we showed that, for $\mathcal{E}_0 = \mathcal{B}$, (2.3) and (2.4) are satisfied with the ellipsoid given by the set of points x satisfying:

$$\left(x + \frac{w/\|w\|_2}{n+1}\right)^\top \left(\frac{n^2-1}{n^2}\mathrm{I}_n + \frac{2(n+1)}{n^2}\frac{ww^\top}{\|w\|_2^2}\right)\left(x + \frac{w/\|w\|_2}{n+1}\right) \leq 1. \tag{2.7}$$

We consider now an arbitrary ellipsoid $\mathcal{E}_0 = \{x \in \mathbb{R}^n : (x - c_0)^\top H_0^{-1}(x - c_0) \leq 1\}$. Let $\Phi(x) = c_0 + H_0^{1/2}x$, then clearly $\mathcal{E}_0 = \Phi(\mathcal{B})$ and $\{x : w^\top(x - c_0) \leq 0\} = \Phi(\{x : (H_0^{1/2}w)^\top x \leq 0\})$. Thus in this case the image by Φ of the ellipsoid given in (2.7) with w replaced by $H_0^{1/2}w$ will satisfy (2.3) and (2.4). It is easy to see that this corresponds to an ellipsoid defined by

$$c = c_0 - \frac{1}{n+1}\frac{H_0 w}{\sqrt{w^\top H_0 w}},$$

$$H^{-1} = \left(1 - \frac{1}{n^2}\right)H_0^{-1} + \frac{2(n+1)}{n^2}\frac{ww^\top}{w^\top H_0 w}. \tag{2.8}$$

Applying Sherman-Morrison formula to (2.8) one can recover (2.6) which concludes the proof. $\qquad\square$

We describe now the ellipsoid method, which only assumes a separation oracle for the constraint set $\mathcal{X}$ (in particular it can be used to solve the feasibility problem mentioned at the beginning of the chapter). Let $\mathcal{E}_0$ be the Euclidean ball of radius R that contains $\mathcal{X}$, and let c_0 be its center. Denote also $H_0 = R^2 \mathrm{I}_n$. For $t \geq 0$ do the following:

1. If $c_t \notin \mathcal{X}$ then call the separation oracle to obtain a separating hyperplane $w_t \in \mathbb{R}^n$ such that $\mathcal{X} \subset \{x : (x - c_t)^\top w_t \leq 0\}$, otherwise call the first order oracle at c_t to obtain $w_t \in \partial f(c_t)$.

2. Let $\mathcal{E}_{t+1} = \{x : (x - c_{t+1})^\top H_{t+1}^{-1}(x - c_{t+1}) \leq 1\}$ be the ellipsoid given in Lemma 2.3 that contains $\{x \in \mathcal{E}_t : (x - c_t)^\top w_t \leq 0\}$, that is

$$c_{t+1} = c_t - \frac{1}{n+1}\frac{H_t w}{\sqrt{w^\top H_t w}},$$

$$H_{t+1} = \frac{n^2}{n^2-1}\left(H_t - \frac{2}{n+1}\frac{H_t ww^\top H_t}{w^\top H_t w}\right).$$

If stopped after t iterations and if $\{c_1, \ldots, c_t\} \cap \mathcal{X} \neq \emptyset$, then we use the zeroth order oracle to output

$$x_t \in \operatorname*{argmin}_{c \in \{c_1, \ldots, c_t\} \cap \mathcal{X}} f(c_r).$$

The following rate of convergence can be proved with the exact same argument than for Theorem 2.1 (observe that at step t one can remove a point in $\mathcal{X}$ from the current ellipsoid only if $c_t \in \mathcal{X}$).

Theorem 2.4. For $t \geq 2n^2 \log(R/r)$ the ellipsoid method satisfies $\{c_1, \ldots, c_t\} \cap \mathcal{X} \neq \emptyset$ and

$$f(x_t) - \min_{x \in \mathcal{X}} f(x) \leq \frac{2BR}{r} \exp\left(-\frac{t}{2n^2}\right).$$

We observe that the oracle complexity of the ellipsoid method is much worse than the one of the center gravity method, indeed the former needs $O(n^2 \log(1/\varepsilon))$ calls to the oracles while the latter requires only $O(n \log(1/\varepsilon))$ calls. However from a computational point of view the situation is much better: in many cases one can derive an efficient separation oracle, while the center of gravity method is basically always intractable. This is for instance the case in the context of LPs and SDPs: with the notation of Section 1.5 the computational complexity of the separation oracle for LPs is $O(mn)$ while for SDPs it is $O(\max(m, n)n^2)$ (we use the fact that the spectral decomposition of a matrix can be done in $O(n^3)$ operations). This gives an overall complexity of $O(\max(m, n)n^3 \log(1/\varepsilon))$ for LPs and $O(\max(m, n^2)n^6 \log(1/\varepsilon))$ for SDPs. We note however that the ellipsoid method is almost never used in practice, essentially because the method is too rigid to exploit the potential easiness of real problems (e.g., the volume decrease given by (2.4) is essentially always tight).

2.3 Vaidya's cutting plane method

We focus here on the feasibility problem (it should be clear from the previous sections how to adapt the argument for optimization). We have seen that for the feasibility problem the center of gravity has a $O(n)$ oracle complexity and unclear computational complexity (see

Section 6.7 for more on this), while the ellipsoid method has oracle complexity $O(n^2)$ and computational complexity $O(n^4)$. We describe here the beautiful algorithm of Vaidya [1989, 1996] which has oracle complexity $O(n \log(n))$ and computational complexity $O(n^4)$, thus getting the best of both the center of gravity and the ellipsoid method. In fact the computational complexity can even be improved further, and the recent breakthrough Lee et al. [2015] shows that it can essentially (up to logarithmic factors) be brought down to $O(n^3)$.

This section, while giving a fundamental algorithm, should probably be skipped on a first reading. In particular we use several concepts from the theory of interior point methods which are described in Section 5.3.

2.3.1 The volumetric barrier

Let $A \in \mathbb{R}^{m \times n}$ where the ith row is $a_i \in \mathbb{R}^n$, and let $b \in \mathbb{R}^m$. We consider the logarithmic barrier F for the polytope $\{x \in \mathbb{R}^n : Ax > b\}$ defined by

$$F(x) = -\sum_{i=1}^{m} \log(a_i^\top x - b_i).$$

We also consider the volumetric barrier v defined by

$$v(x) = \frac{1}{2} \text{logdet}(\nabla^2 F(x)).$$

The intuition is clear: $v(x)$ is equal to the logarithm of the inverse volume of the Dikin ellipsoid (for the logarithmic barrier) at x. It will be useful to spell out the hessian of the logarithmic barrier:

$$\nabla^2 F(x) = \sum_{i=1}^{m} \frac{a_i a_i^\top}{(a_i^\top x - b_i)^2}.$$

Introducing the leverage score

$$\sigma_i(x) = \frac{(\nabla^2 F(x))^{-1}[a_i, a_i]}{(a_i^\top x - b_i)^2},$$

one can easily verify that

$$\nabla v(x) = -\sum_{i=1}^{m} \sigma_i(x) \frac{a_i}{a_i^\top x - b_i}, \tag{2.9}$$

and

$$\nabla^2 v(x) \succeq \sum_{i=1}^{m} \sigma_i(x) \frac{a_i a_i^\top}{(a_i^\top x - b_i)^2} =: Q(x). \tag{2.10}$$

2.3.2 Vaidya's algorithm

We fix $\varepsilon \leq 0.006$ a small constant to be specified later. Vaidya's algorithm produces a sequence of pairs $(A^{(t)}, b^{(t)}) \in \mathbb{R}^{m_t \times n} \times \mathbb{R}^{m_t}$ such that the corresponding polytope contains the convex set of interest. The initial polytope defined by $(A^{(0)}, b^{(0)})$ is a simplex (in particular $m_0 = n+1$). For $t \geq 0$ we let x_t be the minimizer of the volumetric barrier v_t of the polytope given by $(A^{(t)}, b^{(t)})$, and $(\sigma_i^{(t)})_{i \in [m_t]}$ the leverage scores (associated to v_t) at the point x_t. We also denote F_t for the logarithmic barrier given by $(A^{(t)}, b^{(t)})$. The next polytope $(A^{(t+1)}, b^{(t+1)})$ is defined by either adding or removing a constraint to the current polytope:

1. If for some $i \in [m_t]$ one has $\sigma_i^{(t)} = \min_{j \in [m_t]} \sigma_j^{(t)} < \varepsilon$, then $(A^{(t+1)}, b^{(t+1)})$ is defined by removing the ith row in $(A^{(t)}, b^{(t)})$ (in particular $m_{t+1} = m_t - 1$).

2. Otherwise let $c^{(t)}$ be the vector given by the separation oracle queried at x_t, and $\beta^{(t)} \in \mathbb{R}$ be chosen so that

   $$\frac{(\nabla^2 F_t(x_t))^{-1}[c^{(t)}, c^{(t)}]}{(x_t^\top c^{(t)} - \beta^{(t)})^2} = \frac{1}{5}\sqrt{\varepsilon}.$$

   Then we define $(A^{(t+1)}, b^{(t+1)})$ by adding to $(A^{(t)}, b^{(t)})$ the row given by $(c^{(t)}, \beta^{(t)})$ (in particular $m_{t+1} = m_t + 1$).

It can be shown that the volumetric barrier is a self-concordant barrier, and thus it can be efficiently minimized with Newton's method. In fact it is enough to do *one step* of Newton's method on v_t initialized at x_{t-1}, see Vaidya [1989, 1996] for more details on this.

2.3.3 Analysis of Vaidya's method

The construction of Vaidya's method is based on a precise understanding of how the volumetric barrier changes when one adds or removes

a constraint to the polytope. This understanding is derived in Section 2.3.4. In particular we obtain the following two key inequalities: If case 1 happens at iteration t then

$$v_{t+1}(x_{t+1}) - v_t(x_t) \geq -\varepsilon, \tag{2.11}$$

while if case 2 happens then

$$v_{t+1}(x_{t+1}) - v_t(x_t) \geq \frac{1}{20}\sqrt{\varepsilon}. \tag{2.12}$$

We show now how these inequalities imply that Vaidya's method stops after $O(n \log(nR/r))$ steps. First we claim that after $2t$ iterations, case 2 must have happened at least $t - 1$ times. Indeed suppose that at iteration $2t - 1$, case 2 has happened $t - 2$ times; then $\nabla^2 F(x)$ is singular and the leverage scores are infinite, so case 2 must happen at iteration $2t$. Combining this claim with the two inequalities above we obtain:

$$v_{2t}(x_{2t}) \geq v_0(x_0) + \frac{t-1}{20}\sqrt{\varepsilon} - (t+1)\varepsilon \geq \frac{t}{50}\varepsilon - 1 + v_0(x_0).$$

The key point now is to recall that by definition one has $v(x) = -\log \mathrm{vol}(\mathcal{E}(x, 1))$ where $\mathcal{E}(x, r) = \{y : \nabla F^2(x)[y - x, y - x] \leq r^2\}$ is the Dikin ellipsoid centered at x and of radius r. Moreover the logarithmic barrier F of a polytope with m constraints is m-self-concordant, which implies that the polytope is included in the Dikin ellipsoid $\mathcal{E}(z, 2m)$ where z is the minimizer of F (see [Theorem 4.2.6., Nesterov [2004a]]). The volume of $\mathcal{E}(z, 2m)$ is equal to $(2m)^n \exp(-v(z))$, which is thus always an upper bound on the volume of the polytope. Combining this with the above display we just proved that at iteration $2k$ the volume of the current polytope is at most

$$\exp\left(n \log(2m_{2t}) + 1 - v_0(x_0) - \frac{t}{50}\varepsilon\right).$$

Since $\mathcal{E}(x, 1)$ is always included in the polytope we have that $-v_0(x_0)$ is at most the logarithm of the volume of the initial polytope which is $O(n \log(R))$. This clearly concludes the proof as the procedure will necessarily stop when the volume is below $\exp(n \log(r))$ (we also used the trivial bound $m_t \leq n + 1 + t$).

2.3.4   Constraints and the volumetric barrier

We want to understand the effect on the volumetric barrier of addition/deletion of constraints to the polytope. Let $c \in \mathbb{R}^n$, $\beta \in \mathbb{R}$, and consider the logarithmic barrier $\widetilde{F}$ and the volumetric barrier $\widetilde{v}$ corresponding to the matrix $\widetilde{A} \in \mathbb{R}^{(m+1)\times n}$ and the vector $\widetilde{b} \in \mathbb{R}^{m+1}$ which are respectively the concatenation of A and c, and the concatenation of b and β. Let x^* and $\widetilde{x}^*$ be the minimizer of respectively v and $\widetilde{v}$. We recall the definition of leverage scores, for $i \in [m+1]$, where $a_{m+1} = c$ and $b_{m+1} = \beta$,

$$\sigma_i(x) = \frac{(\nabla^2 F(x))^{-1}[a_i, a_i]}{(a_i^\top x - b_i)^2}, \text{ and } \widetilde{\sigma}_i(x) = \frac{(\nabla^2 \widetilde{F}(x))^{-1}[a_i, a_i]}{(a_i^\top x - b_i)^2}.$$

The leverage scores σ_i and $\widetilde{\sigma}_i$ are closely related:

Lemma 2.5. One has for any $i \in [m+1]$,

$$\frac{\widetilde{\sigma}_{m+1}(x)}{1 - \widetilde{\sigma}_{m+1}(x)} \geq \sigma_i(x) \geq \widetilde{\sigma}_i(x) \geq (1 - \sigma_{m+1}(x))\sigma_i(x).$$

Proof. First we observe that by Sherman-Morrison's formula $(A + uv^\top)^{-1} = A^{-1} - \frac{A^{-1}uv^\top A^{-1}}{1 + A^{-1}[u,v]}$ one has

$$(\nabla^2 \widetilde{F}(x))^{-1} = (\nabla^2 F(x))^{-1} - \frac{(\nabla^2 F(x))^{-1}cc^\top(\nabla^2 F(x))^{-1}}{(c^\top x - \beta)^2 + (\nabla^2 F(x))^{-1}[c,c]}, \quad (2.13)$$

This immediately proves $\widetilde{\sigma}_i(x) \leq \sigma_i(x)$. It also implies the inequality $\widetilde{\sigma}_i(x) \geq (1-\sigma_{m+1}(x))\sigma_i(x)$ thanks the following fact: $A - \frac{Auu^\top A}{1+A[u,u]} \succeq (1 - A[u,u])A$. For the last inequality we use that $A + \frac{Auu^\top A}{1+A[u,u]} \preceq \frac{1}{1-A[u,u]}A$ together with

$$(\nabla^2 F(x))^{-1} = (\nabla^2 \widetilde{F}(x))^{-1} + \frac{(\nabla^2 \widetilde{F}(x))^{-1}cc^\top(\nabla^2 \widetilde{F}(x))^{-1}}{(c^\top x - \beta)^2 - (\nabla^2 \widetilde{F}(x))^{-1}[c,c]}.$$

$\square$

We now assume the following key result, which was first proven by Vaidya. To put the statement in context recall that for a self-concordant barrier f the suboptimality gap $f(x) - \min f$ is intimately related to the Newton decrement $\|\nabla f(x)\|_{(\nabla^2 f(x))^{-1}}$. Vaidya's inequality gives a

similar claim for the volumetric barrier. We use the version given in [Theorem 2.6, Anstreicher [1998]] which has slightly better numerical constants than the original bound. Recall also the definition of Q from (2.10).

Theorem 2.6. Let $\lambda(x) = \|\nabla v(x)\|_{Q(x)^{-1}}$ be an approximate Newton decrement, $\varepsilon = \min_{i\in[m]} \sigma_i(x)$, and assume that $\lambda(x)^2 \leq \frac{2\sqrt{\varepsilon}-\varepsilon}{36}$. Then

$$v(x) - v(x^*) \leq 2\lambda(x)^2.$$

We also denote $\widetilde{\lambda}$ for the approximate Newton decrement of $\widetilde{v}$. The goal for the rest of the section is to prove the following theorem which gives the precise understanding of the volumetric barrier we were looking for.

Theorem 2.7. Let $\varepsilon := \min_{i\in[m]} \sigma_i(x^*)$, $\delta := \sigma_{m+1}(x^*)/\sqrt{\varepsilon}$ and assume that $\frac{\left(\delta\sqrt{\varepsilon}+\sqrt{\delta^3\sqrt{\varepsilon}}\right)^2}{1-\delta\sqrt{\varepsilon}} < \frac{2\sqrt{\varepsilon}-\varepsilon}{36}$. Then one has

$$\widetilde{v}(\widetilde{x}^*) - v(x^*) \geq \frac{1}{2}\log(1+\delta\sqrt{\varepsilon}) - 2\frac{\left(\delta\sqrt{\varepsilon}+\sqrt{\delta^3\sqrt{\varepsilon}}\right)^2}{1-\delta\sqrt{\varepsilon}}. \qquad (2.14)$$

On the other hand assuming that $\widetilde{\sigma}_{m+1}(\widetilde{x}^*) = \min_{i\in[m+1]} \widetilde{\sigma}_i(\widetilde{x}^*) =: \varepsilon$ and that $\varepsilon \leq 1/4$, one has

$$\widetilde{v}(\widetilde{x}^*) - v(x^*) \leq -\frac{1}{2}\log(1-\varepsilon) + \frac{8\varepsilon^2}{(1-\varepsilon)^2}. \qquad (2.15)$$

Before going into the proof let us see briefly how Theorem 2.7 give the two inequalities stated at the beginning of Section 2.3.3. To prove (2.12) we use (2.14) with $\delta = 1/5$ and $\varepsilon \leq 0.006$, and we observe that in this case the right hand side of (2.14) is lower bounded by $\frac{1}{20}\sqrt{\varepsilon}$. On the other hand to prove (2.11) we use (2.15), and we observe that for $\varepsilon \leq 0.006$ the right hand side of (2.15) is upper bounded by ε.

Proof. We start with the proof of (2.14). First observe that by factoring

$(\nabla^2 F(x))^{1/2}$ on the left and on the right of $\nabla^2 \widetilde{F}(x)$ one obtains

$$
\det(\nabla^2 \widetilde{F}(x))
$$

$$
= \det\left(\nabla^2 F(x) + \frac{cc^\top}{(c^\top x - \beta)^2}\right)
$$

$$
= \det(\nabla^2 F(x))\det\left(\mathrm{I}_n + \frac{(\nabla^2 F(x))^{-1/2}cc^\top(\nabla^2 F(x))^{-1/2}}{(c^\top x - \beta)^2}\right)
$$

$$
= \det(\nabla^2 F(x))(1 + \sigma_{m+1}(x)),
$$

and thus

$$
\widetilde{v}(x) = v(x) + \frac{1}{2}\log(1 + \sigma_{m+1}(x)).
$$

In particular we have

$$
\widetilde{v}(\widetilde{x}^*) - v(x^*) = \frac{1}{2}\log(1 + \sigma_{m+1}(x^*)) - (\widetilde{v}(x^*) - \widetilde{v}(\widetilde{x}^*)).
$$

To bound the suboptimality gap of x^* in $\widetilde{v}$ we will invoke Theorem 2.6 and thus we have to upper bound the approximate Newton decrement $\widetilde{\lambda}$. Using [(2.16), Lemma 2.8] below one has

$$
\widetilde{\lambda}(x^*)^2 \le \frac{\left(\sigma_{m+1}(x^*) + \sqrt{\frac{\sigma_{m+1}^3(x^*)}{\min_{i \in [m]}\sigma_i(x^*)}}\right)^2}{1 - \sigma_{m+1}(x^*)} = \frac{\left(\delta\sqrt{\varepsilon} + \sqrt{\delta^3\sqrt{\varepsilon}}\right)^2}{1 - \delta\sqrt{\varepsilon}}.
$$

This concludes the proof of (2.14).

We now turn to the proof of (2.15). Following the same steps as above we immediately obtain

$$
\begin{aligned}
\widetilde{v}(\widetilde{x}^*) - v(x^*) &= \widetilde{v}(\widetilde{x}^*) - v(\widetilde{x}^*) + v(\widetilde{x}^*) - v(x^*) \\
&= -\frac{1}{2}\log(1 - \widetilde{\sigma}_{m+1}(\widetilde{x}^*)) + v(\widetilde{x}^*) - v(x^*).
\end{aligned}
$$

To invoke Theorem 2.6 it remains to upper bound $\lambda(\widetilde{x}^*)$. Using [(2.17), Lemma 2.8] below one has

$$
\lambda(\widetilde{x}^*) \le \frac{2\,\widetilde{\sigma}_{m+1}(\widetilde{x}^*)}{1 - \widetilde{\sigma}_{m+1}(\widetilde{x}^*)}.
$$

We can apply Theorem 2.6 since the assumption $\varepsilon \le 1/4$ implies that $\left(\frac{2\varepsilon}{1-\varepsilon}\right)^2 \le \frac{2\sqrt{\varepsilon}-\varepsilon}{36}$. This concludes the proof of (2.15). $\qquad\square$

Lemma 2.8. One has

$$\sqrt{1 - \sigma_{m+1}(x)} \, \widetilde{\lambda}(x) \leq \|\nabla v(x)\|_{Q(x)^{-1}} + \sigma_{m+1}(x) + \sqrt{\frac{\sigma_{m+1}^3(x)}{\min_{i \in [m]} \sigma_i(x)}}.$$
$$(2.16)$$

Furthermore if $\widetilde{\sigma}_{m+1}(x) = \min_{i \in [m+1]} \widetilde{\sigma}_i(x)$ then one also has

$$\lambda(x) \leq \|\nabla \widetilde{v}(x)\|_{Q(x)^{-1}} + \frac{2 \, \widetilde{\sigma}_{m+1}(x)}{1 - \widetilde{\sigma}_{m+1}(x)}. \qquad (2.17)$$

Proof. We start with the proof of (2.16). First observe that by Lemma 2.5 one has $\widetilde{Q}(x) \succeq (1 - \sigma_{m+1}(x))Q(x)$ and thus by definition of the Newton decrement

$$\widetilde{\lambda}(x) = \|\nabla \widetilde{v}(x)\|_{\widetilde{Q}(x)^{-1}} \leq \frac{\|\nabla \widetilde{v}(x)\|_{Q(x)^{-1}}}{\sqrt{1 - \sigma_{m+1}(x)}}.$$

Next observe that (recall (2.9))

$$\nabla \widetilde{v}(x) = \nabla v(x) + \sum_{i=1}^{m} (\sigma_i(x) - \widetilde{\sigma}_i(x)) \frac{a_i}{a_i^\top x - b_i} - \widetilde{\sigma}_{m+1}(x) \frac{c}{c^\top x - \beta}.$$

We now use that $Q(x) \succeq (\min_{i \in [m]} \sigma_i(x))\nabla^2 F(x)$ to obtain

$$\left\| \widetilde{\sigma}_{m+1}(x) \frac{c}{c^\top x - \beta} \right\|_{Q(x)^{-1}}^2 \leq \frac{\widetilde{\sigma}_{m+1}^2(x)\sigma_{m+1}(x)}{\min_{i \in [m]} \sigma_i(x)}.$$

By Lemma 2.5 one has $\widetilde{\sigma}_{m+1}(x) \leq \sigma_{m+1}(x)$ and thus we see that it only remains to prove

$$\left\| \sum_{i=1}^{m} (\sigma_i(x) - \widetilde{\sigma}_i(x)) \frac{a_i}{a_i^\top x - b_i} \right\|_{Q(x)^{-1}}^2 \leq \sigma_{m+1}^2(x).$$

The above inequality follows from a beautiful calculation of Vaidya (see [Lemma 12, Vaidya [1996]]), starting from the identity

$$\sigma_i(x) - \widetilde{\sigma}_i(x) = \frac{((\nabla^2 F(x))^{-1}[a_i, c])^2}{((c^\top x - \beta)^2 + (\nabla^2 F(x))^{-1}[c, c])(a_i^\top x - b_i)^2},$$

which itself follows from (2.13).

We now turn to the proof of (2.17). Following the same steps as above we immediately obtain

$$\lambda(x) = \|\nabla v(x)\|_{Q(x)^{-1}} \le \|\nabla \widetilde{v}(x)\|_{Q(x)^{-1}} + \sigma_{m+1}(x) + \sqrt{\frac{\widetilde{\sigma}_{m+1}^2(x)\sigma_{m+1}(x)}{\min_{i\in[m]}\sigma_i(x)}}.$$

Using Lemma 2.5 together with the assumption $\widetilde{\sigma}_{m+1}(x) = \min_{i\in[m+1]}\widetilde{\sigma}_i(x)$ yields (2.17), thus concluding the proof. $\square$

2.4 Conjugate gradient

We conclude this chapter with the special case of unconstrained optimization of a convex quadratic function $f(x) = \frac{1}{2}x^\top A x - b^\top x$, where $A \in \mathbb{R}^{n\times n}$ is a positive definite matrix and $b \in \mathbb{R}^n$. This problem, of paramount importance in practice (it is equivalent to solving the linear system $Ax = b$), admits a simple first-order black-box procedure which attains the *exact* optimum x^* in at most n steps. This method, called the *conjugate gradient*, is described and analyzed below. What is written below is taken from [Chapter 5, Nocedal and Wright [2006]].

Let $\langle \cdot, \cdot \rangle_A$ be the inner product on $\mathbb{R}^n$ defined by the positive definite matrix A, that is $\langle x, y \rangle_A = x^\top A y$ (we also denote by $\| \cdot \|_A$ the corresponding norm). For sake of clarity we denote here $\langle \cdot, \cdot \rangle$ for the standard inner product in $\mathbb{R}^n$. Given an orthogonal set $\{p_0, \dots, p_{n-1}\}$ for $\langle \cdot, \cdot \rangle_A$ we will minimize f by sequentially minimizing it along the directions given by this orthogonal set. That is, given $x_0 \in \mathbb{R}^n$, for $t \ge 0$ let

$$x_{t+1} := \operatorname*{argmin}_{x \in \{x_t + \lambda p_t, \ \lambda \in \mathbb{R}\}} f(x). \tag{2.18}$$

Equivalently one can write

$$x_{t+1} = x_t - \langle \nabla f(x_t), p_t \rangle \frac{p_t}{\|p_t\|_A^2}. \tag{2.19}$$

The latter identity follows by differentiating $\lambda \mapsto f(x + \lambda p_t)$, and using that $\nabla f(x) = Ax - b$. We also make an observation that will be useful later, namely that x_{t+1} is the minimizer of f on $x_0 + \operatorname{span}\{p_0, \dots, p_t\}$, or equivalently

$$\langle \nabla f(x_{t+1}), p_i \rangle = 0, \forall \ 0 \le i \le t. \tag{2.20}$$

Equation (2.20) is true by construction for $i = t$, and for $i \leq t - 1$ it follows by induction, assuming (2.20) at $t = 1$ and using the following formula:

$$\nabla f(x_{t+1}) = \nabla f(x_t) - \langle \nabla f(x_t), p_t \rangle \frac{Ap_t}{\|p_t\|_A^2}. \tag{2.21}$$

We now claim that $x_n = x^* = \operatorname{argmin}_{x \in \mathbb{R}^n} f(x)$. It suffices to show that $\langle x_n - x_0, p_t \rangle_A = \langle x^* - x_0, p_t \rangle_A$ for any $t \in \{0, \dots, n-1\}$. Note that $x_n - x_0 = -\sum_{t=0}^{n-1} \langle \nabla f(x_t), p_t \rangle \frac{p_t}{\|p_t\|_A^2}$, and thus using that $x^* = A^{-1}b$,

$$\langle x_n - x_0, p_t \rangle_A = -\langle \nabla f(x_t), p_t \rangle = \langle b - Ax_t, p_t \rangle = \langle x^* - x_t, p_t \rangle_A$$
$$= \langle x^* - x_0, p_t \rangle_A,$$

which concludes the proof of $x_n = x^*$.

In order to have a proper black-box method it remains to describe how to build iteratively the orthogonal set $\{p_0, \dots, p_{n-1}\}$ based only on gradient evaluations of f. A natural guess to obtain a set of orthogonal directions (w.r.t. $\langle \cdot, \cdot \rangle_A$) is to take $p_0 = \nabla f(x_0)$ and for $t \geq 1$,

$$p_t = \nabla f(x_t) - \langle \nabla f(x_t), p_{t-1} \rangle_A \frac{p_{t-1}}{\|p_{t-1}\|_A^2}. \tag{2.22}$$

Let us first verify by induction on $t \in [n-1]$ that for any $i \in \{0, \dots, t-2\}$, $\langle p_t, p_i \rangle_A = 0$ (observe that for $i = t-1$ this is true by construction of p_t). Using the induction hypothesis one can see that it is enough to show $\langle \nabla f(x_t), p_i \rangle_A = 0$ for any $i \in \{0, \dots, t-2\}$, which we prove now. First observe that by induction one easily obtains $Ap_i \in \operatorname{span}\{p_0, \dots, p_{i+1}\}$ from (2.21) and (2.22). Using this fact together with $\langle \nabla f(x_t), p_i \rangle_A = \langle \nabla f(x_t), Ap_i \rangle$ and (2.20) thus concludes the proof of orthogonality of the set $\{p_0, \dots, p_{n-1}\}$.

We still have to show that (2.22) can be written by making only reference to the gradients of f at previous points. Recall that x_{t+1} is the minimizer of f on $x_0 + \operatorname{span}\{p_0, \dots, p_t\}$, and thus given the form of p_t we also have that x_{t+1} is the minimizer of f on $x_0 + \operatorname{span}\{\nabla f(x_0), \dots, \nabla f(x_t)\}$ (in some sense the conjugate gradient is the *optimal* first order method for convex quadratic functions). In particular one has $\langle \nabla f(x_{t+1}), \nabla f(x_t) \rangle = 0$. This fact, together with

the orthogonality of the set $\{p_t\}$ and (2.21), imply that

$$\frac{\langle \nabla f(x_{t+1}), p_t \rangle_A}{\|p_t\|_A^2} = \langle \nabla f(x_{t+1}), \frac{Ap_t}{\|p_t\|_A^2} \rangle = -\frac{\langle \nabla f(x_{t+1}), \nabla f(x_{t+1}) \rangle}{\langle \nabla f(x_t), p_t \rangle}.$$

Furthermore using the definition (2.22) and $\langle \nabla f(x_t), p_{t-1} \rangle = 0$ one also has

$$\langle \nabla f(x_t), p_t \rangle = \langle \nabla f(x_t), \nabla f(x_t) \rangle.$$

Thus we arrive at the following rewriting of the (linear) conjugate gradient algorithm, where we recall that x_0 is some fixed starting point and $p_0 = \nabla f(x_0)$,

$$x_{t+1} = \underset{x \in \{x_t + \lambda p_t, \ \lambda \in \mathbb{R}\}}{\mathrm{argmin}} f(x), \tag{2.23}$$

$$p_{t+1} = \nabla f(x_{t+1}) + \frac{\langle \nabla f(x_{t+1}), \nabla f(x_{t+1}) \rangle}{\langle \nabla f(x_t), \nabla f(x_t) \rangle} p_t. \tag{2.24}$$

Observe that the algorithm defined by (2.23) and (2.24) makes sense for an arbitary convex function, in which case it is called the *non-linear conjugate gradient*. There are many variants of the non-linear conjugate gradient, and the above form is known as the Fletcher-âĂŞReeves method. Another popular version in practice is the Polak-Ribière method which is based on the fact that for the general non-quadratic case one does not necessarily have $\langle \nabla f(x_{t+1}), \nabla f(x_t) \rangle = 0$, and thus one replaces (2.24) by

$$p_{t+1} = \nabla f(x_{t+1}) + \frac{\langle \nabla f(x_{t+1}) - \nabla f(x_t), \nabla f(x_{t+1}) \rangle}{\langle \nabla f(x_t), \nabla f(x_t) \rangle} p_t.$$

We refer to Nocedal and Wright [2006] for more details about these algorithms, as well as for advices on how to deal with the line search in (2.23).

Finally we also note that the linear conjugate gradient method can often attain an approximate solution in much fewer than n steps. More precisely, denoting κ for the condition number of A (that is the ratio of the largest eigenvalue to the smallest eigenvalue of A), one can show that linear conjugate gradient attains an ε optimal point in a number of iterations of order $\sqrt{\kappa} \log(1/\varepsilon)$. The next chapter will demistify this

convergence rate, and in particular we will see that (i) this is the optimal rate among first order methods, and (ii) there is a way to generalize this rate to non-quadratic convex functions (though the algorithm will have to be modified).

3

Dimension-free convex optimization

We investigate here <u>variants</u> of the *gradient descent* scheme. This iterative algorithm, which can be traced back to Cauchy [1847], is the simplest strategy to minimize a differentiable function f on $\mathbb{R}^n$. Starting at some initial point $x_1 \in \mathbb{R}^n$ it iterates the following equation:

$$x_{t+1} = x_t - \eta \nabla f(x_t), \tag{3.1}$$

where $\eta > 0$ is a fixed step-size parameter. The rationale behind (3.1) is to make a small step in the direction that minimizes the local first order Taylor approximation of f (also known as the steepest descent direction).

As we shall see, methods of the type (3.1) can obtain an oracle complexity *independent of the dimension*[1]. This feature makes them particularly attractive for optimization in very high dimension.

Apart from Section 3.3, in this chapter $\|\cdot\|$ denotes the Euclidean norm. The set of constraints $\mathcal{X} \subset \mathbb{R}^n$ is assumed to be compact and

---

[1]Of course the computational complexity remains at least linear in the dimension since one needs to manipulate gradients.
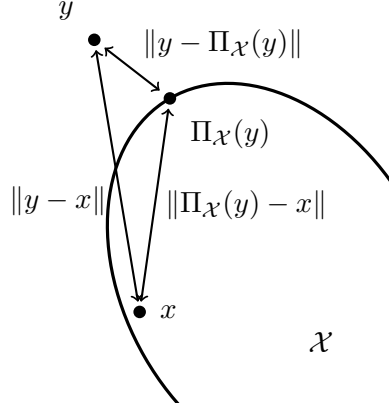
Figure 3.1: Illustration of Lemma 3.1.

convex. We define the projection operator $\Pi_\mathcal{X}$ on $\mathcal{X}$ by

$$\Pi_\mathcal{X}(x) = \operatorname*{argmin}_{y \in \mathcal{X}} \|x - y\|.$$

The following lemma will prove to be useful in our study. It is an easy corollary of Proposition 1.3, see also Figure 3.1.

Lemma 3.1. Let $x \in \mathcal{X}$ and $y \in \mathbb{R}^n$, then

$$(\Pi_\mathcal{X}(y) - x)^\top (\Pi_\mathcal{X}(y) - y) \le 0,$$

which also implies $\|\Pi_\mathcal{X}(y) - x\|^2 + \|y - \Pi_\mathcal{X}(y)\|^2 \le \|y - x\|^2$.

Unless specified otherwise all the proofs in this chapter are taken from Nesterov [2004a] (with slight simplification in some cases).

3.1 Projected subgradient descent for Lipschitz functions

In this section we assume that $\mathcal{X}$ is contained in an Euclidean ball centered at $x_1 \in \mathcal{X}$ and of radius R. Furthermore we assume that f is such that for any $x \in \mathcal{X}$ and any $g \in \partial f(x)$ (we assume $\partial f(x) \ne \emptyset$),
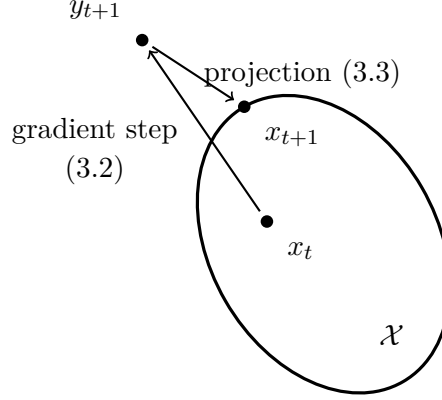
Figure 3.2: Illustration of the projected subgradient descent method.

one has $\|g\| \leq L$. Note that by the subgradient inequality and Cauchy-Schwarz this implies that f is L-Lipschitz on $\mathcal{X}$, that is $|f(x) - f(y)| \leq L\|x - y\|$.

In this context we make two modifications to the basic gradient descent (3.1). First, obviously, we replace the gradient $\nabla f(x)$ (which may not exist) by a subgradient $g \in \partial f(x)$. Secondly, and more importantly, we make sure that the updated point lies in $\mathcal{X}$ by projecting back (if necessary) onto it. This gives the *projected subgradient descent* algorithm[2] which iterates the following equations for $t \geq 1$:

$$y_{t+1} = x_t - \eta g_t, \text{ where } g_t \in \partial f(x_t), \tag{3.2}$$

$$x_{t+1} = \Pi_{\mathcal{X}}(y_{t+1}). \tag{3.3}$$

This procedure is illustrated in Figure 3.2. We prove now a rate of convergence for this method under the above assumptions.

Theorem 3.2. The projected subgradient descent method with $\eta =$

---

[2] In the optimization literature the term "descent" is reserved for methods such that $f(x_{t+1}) \leq f(x_t)$. In that sense the projected subgradient descent is not a descent method.

$\frac{R}{L\sqrt{t}}$ satisfies

$$f\left(\frac{1}{t}\sum_{s=1}^{t} x_s\right) - f(x^*) \leq \frac{RL}{\sqrt{t}}.$$

Proof. Using the definition of subgradients, the definition of the method, and the elementary identity $2a^\top b = \|a\|^2 + \|b\|^2 - \|a - b\|^2$, one obtains

$$
\begin{aligned}
f(x_s) - f(x^*) &\leq g_s^\top (x_s - x^*) \\
&= \frac{1}{\eta}(x_s - y_{s+1})^\top (x_s - x^*) \\
&= \frac{1}{2\eta}\left(\|x_s - x^*\|^2 + \|x_s - y_{s+1}\|^2 - \|y_{s+1} - x^*\|^2\right) \\
&= \frac{1}{2\eta}\left(\|x_s - x^*\|^2 - \|y_{s+1} - x^*\|^2\right) + \frac{\eta}{2}\|g_s\|^2.
\end{aligned}
$$

Now note that $\|g_s\| \leq L$, and furthermore by Lemma 3.1

$$\|y_{s+1} - x^*\| \geq \|x_{s+1} - x^*\|.$$

Summing the resulting inequality over s, and using that $\|x_1 - x^*\| \leq R$ yield

$$\sum_{s=1}^{t}(f(x_s) - f(x^*)) \leq \frac{R^2}{2\eta} + \frac{\eta L^2 t}{2}.$$

Plugging in the value of η directly gives the statement (recall that by convexity $f((1/t)\sum_{s=1}^{t} x_s) \leq \frac{1}{t}\sum_{s=1}^{t} f(x_s)$). $\qquad\square$

We will show in Section 3.5 that the rate given in Theorem 3.2 is unimprovable from a black-box perspective. Thus to reach an ε-optimal point one needs $\Theta(1/\varepsilon^2)$ calls to the oracle. In some sense this is an astonishing result as this complexity is independent[3] of the ambient dimension n. On the other hand this is also quite disappointing compared to the scaling in $\log(1/\varepsilon)$ of the center of gravity and ellipsoid method of Chapter 2. To put it differently with gradient descent one could hope to reach a reasonable accuracy in very high dimension, while with the ellipsoid method one can reach very high accuracy in

---

[3]Observe however that the quantities R and L may dependent on the dimension, see Chapter 4 for more on this.

reasonably small dimension. A major task in the following sections will be to explore more restrictive assumptions on the function to be optimized in order to have the best of both worlds, that is an oracle complexity independent of the dimension and with a scaling in $\log(1/\varepsilon)$.

The computational bottleneck of the projected subgradient descent is often the projection step (3.3) which is a convex optimization problem by itself. In some cases this problem may admit an analytical solution (think of $\mathcal{X}$ being an Euclidean ball), or an easy and fast combinatorial algorithm to solve it (this is the case for $\mathcal{X}$ being an ℓ_1-ball, see Maculan and de Paula [1989]). We will see in Section 3.3 a projection-free algorithm which operates under an extra assumption of smoothness on the function to be optimized.

Finally we observe that the step-size recommended by Theorem 3.2 depends on the number of iterations to be performed. In practice this may be an undesirable feature. However using a time-varying step size of the form $\eta_s = \frac{R}{L\sqrt{s}}$ one can prove the same rate up to a $\log t$ factor. In any case these step sizes are very small, which is the reason for the slow convergence. In the next section we will see that by assuming *smoothness* in the function f one can afford to be much more aggressive. Indeed in this case, as one approaches the optimum the size of the gradients themselves will go to 0, resulting in a sort of "auto-tuning" of the step sizes which does not happen for an arbitrary convex function.

3.2   Gradient descent for smooth functions

We say that a continuously differentiable function f is β-smooth if the gradient ∇f is β-Lipschitz, that is

$$\|\nabla f(x) - \nabla f(y)\| \leq \beta \|x - y\|.$$

Note that if f is twice differentiable then this is equivalent to the eigenvalues of the Hessians being smaller than β. In this section we explore potential improvements in the rate of convergence under such a smoothness assumption. In order to avoid technicalities we consider first the unconstrained situation, where f is a convex and β-smooth function on $\mathbb{R}^n$. The next theorem shows that *gradient descent*, which iterates

$x_{t+1} = x_t - \eta \nabla f(x_t)$, attains a much faster rate in this situation than in the non-smooth case of the previous section.

Theorem 3.3. Let f be convex and β-smooth on $\mathbb{R}^n$. Then gradient descent with $\eta = \frac{1}{\beta}$ satisfies

$$f(x_t) - f(x^*) \leq \frac{2\beta \|x_1 - x^*\|^2}{t-1}.$$

Before embarking on the proof we state a few properties of smooth convex functions.

Lemma 3.4. Let f be a β-smooth function on $\mathbb{R}^n$. Then for any $x, y \in \mathbb{R}^n$, one has

$$|f(x) - f(y) - \nabla f(y)^\top (x-y)| \leq \frac{\beta}{2} \|x - y\|^2.$$

Proof. We represent $f(x) - f(y)$ as an integral, apply Cauchy-Schwarz and then β-smoothness:

$$
\begin{aligned}
&|f(x) - f(y) - \nabla f(y)^\top (x-y)| \\
&= \left| \int_0^1 \nabla f(y + t(x-y))^\top (x-y) dt - \nabla f(y)^\top (x-y) \right| \\
&\leq \int_0^1 \|\nabla f(y + t(x-y)) - \nabla f(y)\| \cdot \|x-y\| dt \\
&\leq \int_0^1 \beta t \|x-y\|^2 dt \\
&= \frac{\beta}{2} \|x-y\|^2.
\end{aligned}
$$

$\square$

In particular this lemma shows that if f is convex and β-smooth, then for any $x, y \in \mathbb{R}^n$, one has

$$0 \leq f(x) - f(y) - \nabla f(y)^\top (x-y) \leq \frac{\beta}{2} \|x-y\|^2. \qquad (3.4)$$

This gives in particular the following important inequality to evaluate the improvement in one step of gradient descent:

$$f\left(x - \frac{1}{\beta} \nabla f(x)\right) - f(x) \leq -\frac{1}{2\beta} \|\nabla f(x)\|^2. \qquad (3.5)$$

The next lemma, which improves the basic inequality for subgradients under the smoothness assumption, shows that in fact f is convex and β-smooth if and only if (3.4) holds true. In the literature (3.4) is often used as a definition of smooth convex functions.

Lemma 3.5. Let f be such that (3.4) holds true. Then for any $x, y \in \mathbb{R}^n$, one has

$$f(x) - f(y) \leq \nabla f(x)^\top (x - y) - \frac{1}{2\beta} \|\nabla f(x) - \nabla f(y)\|^2.$$

Proof. Let $z = y - \frac{1}{\beta}(\nabla f(y) - \nabla f(x))$. Then one has

$$
\begin{aligned}
&f(x) - f(y) \\
&= f(x) - f(z) + f(z) - f(y) \\
&\leq \nabla f(x)^\top (x - z) + \nabla f(y)^\top (z - y) + \frac{\beta}{2} \|z - y\|^2 \\
&= \nabla f(x)^\top (x - y) + (\nabla f(x) - \nabla f(y))^\top (y - z) + \frac{1}{2\beta} \|\nabla f(x) - \nabla f(y)\|^2 \\
&= \nabla f(x)^\top (x - y) - \frac{1}{2\beta} \|\nabla f(x) - \nabla f(y)\|^2.
\end{aligned}
$$

$\square$

We can now prove Theorem 3.3

Proof. Using (3.5) and the definition of the method one has

$$f(x_{s+1}) - f(x_s) \leq -\frac{1}{2\beta} \|\nabla f(x_s)\|^2.$$

In particular, denoting $\delta_s = f(x_s) - f(x^*)$, this shows:

$$\delta_{s+1} \leq \delta_s - \frac{1}{2\beta} \|\nabla f(x_s)\|^2.$$

One also has by convexity

$$\delta_s \leq \nabla f(x_s)^\top (x_s - x^*) \leq \|x_s - x^*\| \cdot \|\nabla f(x_s)\|.$$

We will prove that $\|x_s - x^*\|$ is decreasing with s, which with the two above displays will imply

$$\delta_{s+1} \leq \delta_s - \frac{1}{2\beta \|x_1 - x^*\|^2} \delta_s^2.$$

Let us see how to use this last inequality to conclude the proof. Let $\omega = \frac{1}{2\beta\|x_1 - x^*\|^2}$, then[4]

$$\omega\delta_s^2 + \delta_{s+1} \leq \delta_s \Leftrightarrow \omega\frac{\delta_s}{\delta_{s+1}} + \frac{1}{\delta_s} \leq \frac{1}{\delta_{s+1}} \Rightarrow \frac{1}{\delta_{s+1}} - \frac{1}{\delta_s} \geq \omega \Rightarrow \frac{1}{\delta_t} \geq \omega(t-1).$$

Thus it only remains to show that $\|x_s - x^*\|$ is decreasing with s. Using Lemma 3.5 one immediately gets

$$(\nabla f(x) - \nabla f(y))^\top (x - y) \geq \frac{1}{\beta}\|\nabla f(x) - \nabla f(y)\|^2. \qquad (3.6)$$

We use this as follows (together with $\nabla f(x^*) = 0$)

$$
\begin{aligned}
\|x_{s+1} - x^*\|^2 &= \|x_s - \frac{1}{\beta}\nabla f(x_s) - x^*\|^2 \\
&= \|x_s - x^*\|^2 - \frac{2}{\beta}\nabla f(x_s)^\top (x_s - x^*) + \frac{1}{\beta^2}\|\nabla f(x_s)\|^2 \\
&\leq \|x_s - x^*\|^2 - \frac{1}{\beta^2}\|\nabla f(x_s)\|^2 \\
&\leq \|x_s - x^*\|^2,
\end{aligned}
$$

which concludes the proof. $\qquad \square$

The constrained case

We now come back to the constrained problem

$$\text{min. } f(x)$$
$$\text{s.t. } x \in \mathcal{X}.$$

Similarly to what we did in Section 3.1 we consider the projected gradient descent algorithm, which iterates $x_{t+1} = \Pi_{\mathcal{X}}(x_t - \eta\nabla f(x_t))$.

The key point in the analysis of gradient descent for unconstrained smooth optimization is that a step of gradient descent started at x will decrease the function value by at least $\frac{1}{2\beta}\|\nabla f(x)\|^2$, see (3.5). In the constrained case we cannot expect that this would still hold true as a step may be cut short by the projection. The next lemma defines the "right" quantity to measure progress in the constrained case.

---

[4]The last step in the sequence of implications can be improved by taking δ_1 into account. Indeed one can easily show with (3.4) that $\delta_1 \leq \frac{1}{4\omega}$. This improves the rate of Theorem 3.3 from $\frac{2\beta\|x_1 - x^*\|^2}{t-1}$ to $\frac{2\beta\|x_1 - x^*\|^2}{t+3}$.

Lemma 3.6. Let $x, y \in \mathcal{X}$, $x^+ = \Pi_{\mathcal{X}}\left(x - \frac{1}{\beta}\nabla f(x)\right)$, and $g_{\mathcal{X}}(x) = \beta(x - x^+)$. Then the following holds true:

$$f(x^+) - f(y) \le g_{\mathcal{X}}(x)^\top (x - y) - \frac{1}{2\beta}\|g_{\mathcal{X}}(x)\|^2.$$

Proof. We first observe that

$$\nabla f(x)^\top (x^+ - y) \le g_{\mathcal{X}}(x)^\top (x^+ - y). \tag{3.7}$$

Indeed the above inequality is equivalent to

$$\left(x^+ - \left(x - \frac{1}{\beta}\nabla f(x)\right)\right)^\top (x^+ - y) \le 0,$$

which follows from Lemma 3.1. Now we use (3.7) as follows to prove the lemma (we also use (3.4) which still holds true in the constrained case)

$$
\begin{aligned}
&f(x^+) - f(y) \\
&= f(x^+) - f(x) + f(x) - f(y) \\
&\le \nabla f(x)^\top (x^+ - x) + \frac{\beta}{2}\|x^+ - x\|^2 + \nabla f(x)^\top (x - y) \\
&= \nabla f(x)^\top (x^+ - y) + \frac{1}{2\beta}\|g_{\mathcal{X}}(x)\|^2 \\
&\le g_{\mathcal{X}}(x)^\top (x^+ - y) + \frac{1}{2\beta}\|g_{\mathcal{X}}(x)\|^2 \\
&= g_{\mathcal{X}}(x)^\top (x - y) - \frac{1}{2\beta}\|g_{\mathcal{X}}(x)\|^2.
\end{aligned}
$$

$\square$

We can now prove the following result.

Theorem 3.7. Let f be convex and β-smooth on $\mathcal{X}$. Then projected gradient descent with $\eta = \frac{1}{\beta}$ satisfies

$$f(x_t) - f(x^*) \le \frac{3\beta\|x_1 - x^*\|^2 + f(x_1) - f(x^*)}{t}.$$

Proof. Lemma 3.6 immediately gives

$$f(x_{s+1}) - f(x_s) \le -\frac{1}{2\beta}\|g_{\mathcal{X}}(x_s)\|^2,$$

and

$$f(x_{s+1}) - f(x^*) \leq \|g_{\mathcal{X}}(x_s)\| \cdot \|x_s - x^*\|.$$

We will prove that $\|x_s - x^*\|$ is decreasing with s, which with the two above displays will imply

$$\delta_{s+1} \leq \delta_s - \frac{1}{2\beta\|x_1 - x^*\|^2}\delta_{s+1}^2.$$

An easy induction shows that

$$\delta_s \leq \frac{3\beta\|x_1 - x^*\|^2 + f(x_1) - f(x^*)}{s}.$$

Thus it only remains to show that $\|x_s - x^*\|$ is decreasing with s. Using Lemma 3.6 one can see that $g_{\mathcal{X}}(x_s)^\top(x_s - x^*) \geq \frac{1}{2\beta}\|g_{\mathcal{X}}(x_s)\|^2$ which implies

$$
\begin{aligned}
\|x_{s+1} - x^*\|^2 &= \left\|x_s - \frac{1}{\beta}g_{\mathcal{X}}(x_s) - x^*\right\|^2 \\
&= \|x_s - x^*\|^2 - \frac{2}{\beta}g_{\mathcal{X}}(x_s)^\top(x_s - x^*) + \frac{1}{\beta^2}\|g_{\mathcal{X}}(x_s)\|^2 \\
&\leq \|x_s - x^*\|^2.
\end{aligned}
$$

$\square$

3.3 Conditional gradient descent, aka Frank-Wolfe

We describe now an alternative algorithm to minimize a smooth convex function f over a compact convex set $\mathcal{X}$. The *conditional gradient descent*, introduced in Frank and Wolfe [1956], performs the following update for $t \geq 1$, where $(\gamma_s)_{s\geq 1}$ is a fixed sequence,

$$y_t \in \operatorname{argmin}_{y\in\mathcal{X}} \nabla f(x_t)^\top y \qquad (3.8)$$

$$x_{t+1} = (1 - \gamma_t)x_t + \gamma_t y_t. \qquad (3.9)$$

In words conditional gradient descent makes a step in the steepest descent direction *given the constraint set $\mathcal{X}$*, see Figure 3.3 for an illustration. From a computational perspective, a key property of this
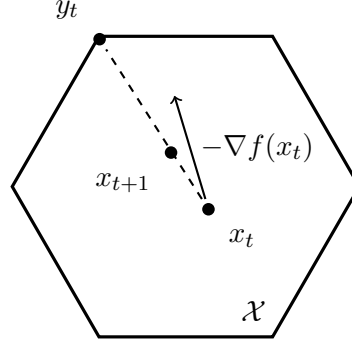
Figure 3.3: Illustration of conditional gradient descent.

scheme is that it replaces the projection step of projected gradient descent by a linear optimization over $\mathcal{X}$, which in some cases can be a much simpler problem.

We now turn to the analysis of this method. A major advantage of conditional gradient descent over projected gradient descent is that the former can adapt to smoothness in an arbitrary norm. Precisely let f be β-smooth in some norm $\|\cdot\|$, that is $\|\nabla f(x) - \nabla f(y)\|_* \leq \beta\|x - y\|$ where the dual norm $\|\cdot\|_*$ is defined as $\|g\|_* = \sup_{x \in \mathbb{R}^n : \|x\| \leq 1} g^\top x$. The following result is extracted from Jaggi [2013] (see also Dunn and Harshbarger [1978]).

Theorem 3.8. Let f be a convex and β-smooth function w.r.t. some norm $\|\cdot\|$, $R = \sup_{x,y \in \mathcal{X}} \|x - y\|$, and $\gamma_s = \frac{2}{s+1}$ for $s \geq 1$. Then for any $t \geq 2$, one has

$$f(x_t) - f(x^*) \leq \frac{2\beta R^2}{t+1}.$$

Proof. The following inequalities hold true, using respectively β-smoothness (it can easily be seen that (3.4) holds true for smoothness in an arbitrary norm), the definition of x_{s+1}, the definition of y_s, and

the convexity of f:

$$
\begin{aligned}
f(x_{s+1}) - f(x_s) &\leq \nabla f(x_s)^\top (x_{s+1} - x_s) + \frac{\beta}{2}\|x_{s+1} - x_s\|^2 \\
&\leq \gamma_s \nabla f(x_s)^\top (y_s - x_s) + \frac{\beta}{2}\gamma_s^2 R^2 \\
&\leq \gamma_s \nabla f(x_s)^\top (x^* - x_s) + \frac{\beta}{2}\gamma_s^2 R^2 \\
&\leq \gamma_s (f(x^*) - f(x_s)) + \frac{\beta}{2}\gamma_s^2 R^2.
\end{aligned}
$$

Rewriting this inequality in terms of $\delta_s = f(x_s) - f(x^*)$ one obtains

$$
\delta_{s+1} \leq (1 - \gamma_s)\delta_s + \frac{\beta}{2}\gamma_s^2 R^2.
$$

A simple induction using that $\gamma_s = \frac{2}{s+1}$ finishes the proof (note that the initialization is done at step 2 with the above inequality yielding $\delta_2 \leq \frac{\beta}{2}R^2$). $\qquad\square$

In addition to being projection-free and "norm-free", the conditional gradient descent satisfies a perhaps even more important property: it produces *sparse iterates*. More precisely consider the situation where $\mathcal{X} \subset \mathbb{R}^n$ is a polytope, that is the convex hull of a finite set of points (these points are called the vertices of $\mathcal{X}$). Then Carathéodory's theorem states that any point $x \in \mathcal{X}$ can be written as a convex combination of at most $n+1$ vertices of $\mathcal{X}$. On the other hand, by definition of the conditional gradient descent, one knows that the tth iterate x_t can be written as a convex combination of t vertices (assuming that x_1 is a vertex). Thanks to the dimension-free rate of convergence one is usually interested in the regime where $t \ll n$, and thus we see that the iterates of conditional gradient descent are very sparse in their vertex representation.

We note an interesting corollary of the sparsity property together with the rate of convergence we proved: smooth functions on the simplex $\{x \in \mathbb{R}_+^n : \sum_{i=1}^n x_i = 1\}$ always admit sparse approximate minimizers. More precisely there must exist a point x with only t non-zero coordinates and such that $f(x) - f(x^*) = O(1/t)$. Clearly this is the best one can hope for in general, as it can be seen with the function

$f(x) = \|x\|_2^2$ since by Cauchy-Schwarz one has $\|x\|_1 \leq \sqrt{\|x\|_0}\|x\|_2$ which implies on the simplex $\|x\|_2^2 \geq 1/\|x\|_0$.

Next we describe an application where the three properties of conditional gradient descent (projection-free, norm-free, and sparse iterates) are critical to develop a computationally efficient procedure.

An application of conditional gradient descent: Least-squares regression with structured sparsity

This example is inspired by Lugosi [2010] (see also Jones [1992]). Consider the problem of approximating a signal $Y \in \mathbb{R}^n$ by a "small" combination of dictionary elements $d_1, \ldots, d_N \in \mathbb{R}^n$. One way to do this is to consider a LASSO type problem in dimension N of the following form (with $\lambda \in \mathbb{R}$ fixed)

$$\min_{x \in \mathbb{R}^N} \left\| Y - \sum_{i=1}^{N} x(i) d_i \right\|_2^2 + \lambda \|x\|_1.$$

Let $D \in \mathbb{R}^{n \times N}$ be the dictionary matrix with i^{th} column given by d_i. Instead of considering the penalized version of the problem one could look at the following constrained problem (with $s \in \mathbb{R}$ fixed) on which we will now focus, see e.g. Friedlander and Tseng [2007],

$$\min_{x \in \mathbb{R}^N} \|Y - Dx\|_2^2 \qquad \Leftrightarrow \qquad \min_{x \in \mathbb{R}^N} \|Y/s - Dx\|_2^2 \qquad (3.10)$$
$$\text{subject to } \|x\|_1 \leq s \qquad\qquad\qquad \text{subject to } \|x\|_1 \leq 1.$$

We make some assumptions on the dictionary. We are interested in situations where the size of the dictionary N can be very large, potentially exponential in the ambient dimension n. Nonetheless we want to restrict our attention to algorithms that run in reasonable time with respect to the ambient dimension n, that is we want polynomial time algorithms in n. Of course in general this is impossible, and we need to assume that the dictionary has some structure that can be exploited. Here we make the assumption that one can do *linear optimization* over the dictionary in polynomial time in n. More precisely we assume that one can solve in time $p(n)$ (where p is polynomial) the following problem for any $y \in \mathbb{R}^n$:

$$\min_{1 \leq i \leq N} y^\top d_i.$$

This assumption is met for many *combinatorial* dictionaries. For instance the dicâŋtioâŋnary eleâŋments could be vecâŋtor of inciâŋdence of spanâŋning trees in some fixed graph, in which case the linâŋear optiâŋmizaâŋtion probâŋlem can be solved with a greedy algorithm.

Finally, for normalization issues, we assume that the ℓ_2-norm of the dictionary elements are controlled by some $m > 0$, that is $\|d_i\|_2 \le m, \forall i \in [N]$.

Our problem of interest (3.10) corresponds to minimizing the function $f(x) = \frac{1}{2}\|Y - Dx\|_2^2$ on the ℓ_1-ball of $\mathbb{R}^N$ in polynomial time in n. At first sight this task may seem completely impossible, indeed one is not even allowed to write down entirely a vector $x \in \mathbb{R}^N$ (since this would take time linear in N). The key property that will save us is that this function admits *sparse minimizers* as we discussed in the previous section, and this will be exploited by the conditional gradient descent method.

First let us study the computational complexity of the tth step of conditional gradient descent. Observe that

$$\nabla f(x) = D^\top (Dx - Y).$$

Now assume that $z_t = Dx_t - Y \in \mathbb{R}^n$ is already computed, then to compute (3.8) one needs to find the coordinate $i_t \in [N]$ that maximizes $|[\nabla f(x_t)](i)|$ which can be done by maximizing $d_i^\top z_t$ and $-d_i^\top z_t$. Thus (3.8) takes time $O(p(n))$. Computing x_{t+1} from x_t and i_t takes time $O(t)$ since $\|x_t\|_0 \le t$, and computing z_{t+1} from z_t and i_t takes time $O(n)$. Thus the overall time complexity of running t steps is (we assume $p(n) = \Omega(n)$)

$$O(tp(n) + t^2). \tag{3.11}$$

To derive a rate of convergence it remains to study the smoothness of f. This can be done as follows:

$$
\begin{aligned}
\|\nabla f(x) - \nabla f(y)\|_\infty &= \|D^\top D(x - y)\|_\infty \\
&= \max_{1 \le i \le N} \left| d_i^\top \left(\sum_{j=1}^N d_j(x(j) - y(j)) \right) \right| \\
&\le m^2 \|x - y\|_1,
\end{aligned}
$$

which means that f is m^2-smooth with respect to the ℓ_1-norm. Thus we get the following rate of convergence:

$$f(x_t) - f(x^*) \leq \frac{8m^2}{t+1}. \tag{3.12}$$

Putting together (3.11) and (3.12) we proved that one can get an ε-optimal solution to (3.10) with a computational effort of $O(m^2 p(n)/\varepsilon + m^4/\varepsilon^2)$ using the conditional gradient descent.

3.4   Strong convexity

We will now discuss another property of convex functions that can significantly speed-up the convergence of first order methods: strong convexity. We say that $f : \mathcal{X} \to \mathbb{R}$ is α-*strongly convex* if it satisfies the following improved subgradient inequality:

$$f(x) - f(y) \leq \nabla f(x)^\top (x - y) - \frac{\alpha}{2}\|x - y\|^2. \tag{3.13}$$

Of course this definition does not require differentiability of the function f, and one can replace $\nabla f(x)$ in the inequality above by $g \in \partial f(x)$. It is immediate to verify that a function f is α-strongly convex if and only if $x \mapsto f(x) - \frac{\alpha}{2}\|x\|^2$ is convex (in particular if f is twice differentiable then the eigenvalues of the Hessians of f have to be larger than α). The strong convexity parameter α is a measure of the *curvature* of f. For instance a linear function has no curvature and hence $\alpha = 0$. On the other hand one can clearly see why a large value of α would lead to a faster rate: in this case a point far from the optimum will have a large gradient, and thus gradient descent will make very big steps when far from the optimum. Of course if the function is non-smooth one still has to be careful and tune the step-sizes to be relatively small, but nonetheless we will be able to improve the oracle complexity from $O(1/\varepsilon^2)$ to $O(1/(\alpha\varepsilon))$. On the other hand with the additional assumption of β-smoothness we will prove that gradient descent with a constant step-size achieves a *linear rate of convergence*, precisely the oracle complexity will be $O(\frac{\beta}{\alpha}\log(1/\varepsilon))$. This achieves the objective we had set after Theorem 3.2: strongly-convex and smooth functions can be optimized in very large dimension and up to very high accuracy.

Before going into the proofs let us discuss another interpretation of strong-convexity and its relation to smoothness. Equation (3.13) can be read as follows: at any point x one can find a (convex) quadratic lower bound $q_x^-(y) = f(x) + \nabla f(x)^\top (y - x) + \frac{\alpha}{2}\|x - y\|^2$ to the function f, i.e. $q_x^-(y) \leq f(y), \forall y \in \mathcal{X}$ (and $q_x^-(x) = f(x)$). On the other hand for β-smoothness (3.4) implies that at any point y one can find a (convex) quadratic upper bound $q_y^+(x) = f(y) + \nabla f(y)^\top (x - y) + \frac{\beta}{2}\|x - y\|^2$ to the function f, i.e. $q_y^+(x) \geq f(x), \forall x \in \mathcal{X}$ (and $q_y^+(y) = f(y)$). Thus in some sense strong convexity is a *dual* assumption to smoothness, and in fact this can be made precise within the framework of Fenchel duality. Also remark that clearly one always has $\beta \geq \alpha$.

3.4.1 Strongly convex and Lipschitz functions

We consider here the projected subgradient descent algorithm with time-varying step size $(\eta_t)_{t \geq 1}$, that is

$$y_{t+1} = x_t - \eta_t g_t, \text{ where } g_t \in \partial f(x_t)$$
$$x_{t+1} = \Pi_{\mathcal{X}}(y_{t+1}).$$

The following result is extracted from Lacoste-Julien et al. [2012].

Theorem 3.9. Let f be α-strongly convex and L-Lipschitz on $\mathcal{X}$. Then projected subgradient descent with $\eta_s = \frac{2}{\alpha(s+1)}$ satisfies

$$f\left(\sum_{s=1}^t \frac{2s}{t(t+1)} x_s\right) - f(x^*) \leq \frac{2L^2}{\alpha(t+1)}.$$

Proof. Coming back to our original analysis of projected subgradient descent in Section 3.1 and using the strong convexity assumption one immediately obtains

$$f(x_s) - f(x^*) \leq \frac{\eta_s}{2} L^2 + \left(\frac{1}{2\eta_s} - \frac{\alpha}{2}\right)\|x_s - x^*\|^2 - \frac{1}{2\eta_s}\|x_{s+1} - x^*\|^2.$$

Multiplying this inequality by s yields

$$s(f(x_s) - f(x^*)) \leq \frac{L^2}{\alpha} + \frac{\alpha}{4}\left(s(s-1)\|x_s - x^*\|^2 - s(s+1)\|x_{s+1} - x^*\|^2\right),$$

Now sum the resulting inequality over $s = 1$ to $s = t$, and apply Jensen's inequality to obtain the claimed statement. $\qquad\square$

3.4.2   Strongly convex and smooth functions

As we will see now, having both strong convexity and smoothness allows for a drastic improvement in the convergence rate. We denote $\kappa = \frac{\beta}{\alpha}$ for the *condition number* of f. The key observation is that Lemma 3.6 can be improved to (with the notation of the lemma):

$$f(x^+) - f(y) \leq g_{\mathcal{X}}(x)^\top (x - y) - \frac{1}{2\beta}\|g_{\mathcal{X}}(x)\|^2 - \frac{\alpha}{2}\|x - y\|^2. \quad (3.14)$$

Theorem 3.10. Let f be α-strongly convex and β-smooth on $\mathcal{X}$. Then projected gradient descent with $\eta = \frac{1}{\beta}$ satisfies for $t \geq 0$,

$$\|x_{t+1} - x^*\|^2 \leq \exp\left(-\frac{t}{\kappa}\right)\|x_1 - x^*\|^2.$$

Proof. Using (3.14) with $y = x^*$ one directly obtains

$$
\begin{aligned}
\|x_{t+1} - x^*\|^2 &= \left\|x_t - \frac{1}{\beta}g_{\mathcal{X}}(x_t) - x^*\right\|^2 \\
&= \|x_t - x^*\|^2 - \frac{2}{\beta}g_{\mathcal{X}}(x_t)^\top(x_t - x^*) + \frac{1}{\beta^2}\|g_{\mathcal{X}}(x_t)\|^2 \\
&\leq \left(1 - \frac{\alpha}{\beta}\right)\|x_t - x^*\|^2 \\
&\leq \left(1 - \frac{\alpha}{\beta}\right)^t \|x_1 - x^*\|^2 \\
&\leq \exp\left(-\frac{t}{\kappa}\right)\|x_1 - x^*\|^2,
\end{aligned}
$$

which concludes the proof. $\square$

We now show that in the unconstrained case one can improve the rate by a constant factor, precisely one can replace κ by $(\kappa + 1)/4$ in the oracle complexity bound by using a larger step size. This is not a spectacular gain but the reasoning is based on an improvement of (3.6) which can be of interest by itself. Note that (3.6) and the lemma to follow are sometimes referred to as *coercivity* of the gradient.

Lemma 3.11. Let f be β-smooth and α-strongly convex on $\mathbb{R}^n$. Then for all $x, y \in \mathbb{R}^n$, one has

$$(\nabla f(x) - \nabla f(y))^\top (x - y) \geq \frac{\alpha\beta}{\beta + \alpha}\|x - y\|^2 + \frac{1}{\beta + \alpha}\|\nabla f(x) - \nabla f(y)\|^2.$$

Proof. Let $\varphi(x) = f(x) - \frac{\alpha}{2}\|x\|^2$. By definition of α-strong convexity one has that φ is convex. Furthermore one can show that φ is $(\beta - \alpha)$-smooth by proving (3.4) (and using that it implies smoothness). Thus using (3.6) one gets

$$(\nabla\varphi(x) - \nabla\varphi(y))^\top(x - y) \geq \frac{1}{\beta - \alpha}\|\nabla\varphi(x) - \nabla\varphi(y)\|^2,$$

which gives the claimed result with straightforward computations. (Note that if $\alpha = \beta$ the smoothness of φ directly implies that $\nabla f(x) - \nabla f(y) = \alpha(x - y)$ which proves the lemma in this case.) $\quad\square$

Theorem 3.12. Let f be β-smooth and α-strongly convex on $\mathbb{R}^n$. Then gradient descent with $\eta = \frac{2}{\alpha+\beta}$ satisfies

$$f(x_{t+1}) - f(x^*) \leq \frac{\beta}{2}\exp\left(-\frac{4t}{\kappa + 1}\right)\|x_1 - x^*\|^2.$$

Proof. First note that by β-smoothness (since $\nabla f(x^*) = 0$) one has

$$f(x_t) - f(x^*) \leq \frac{\beta}{2}\|x_t - x^*\|^2.$$

Now using Lemma 3.11 one obtains

$$
\begin{aligned}
\|x_{t+1} - x^*\|^2 &= \|x_t - \eta\nabla f(x_t) - x^*\|^2 \\
&= \|x_t - x^*\|^2 - 2\eta\nabla f(x_t)^\top(x_t - x^*) + \eta^2\|\nabla f(x_t)\|^2 \\
&\leq \left(1 - 2\frac{\eta\alpha\beta}{\beta + \alpha}\right)\|x_t - x^*\|^2 + \left(\eta^2 - 2\frac{\eta}{\beta + \alpha}\right)\|\nabla f(x_t)\|^2 \\
&= \left(\frac{\kappa - 1}{\kappa + 1}\right)^2\|x_t - x^*\|^2 \\
&\leq \exp\left(-\frac{4t}{\kappa + 1}\right)\|x_1 - x^*\|^2,
\end{aligned}
$$

which concludes the proof. $\quad\square$

3.5 Lower bounds

We prove here various oracle complexity lower bounds. These results first appeared in Nemirovski and Yudin [1983] but we follow here the

simplified presentation of Nesterov [2004a]. In general a black-box procedure is a mapping from "history" to the next query point, that is it maps $(x_1, g_1, \ldots, x_t, g_t)$ (with $g_s \in \partial f(x_s)$) to x_{t+1}. In order to simplify the notation and the argument, throughout the section we make the following assumption on the black-box procedure: $x_1 = 0$ and for any $t \geq 0$, x_{t+1} is in the linear span of $g_1, \ldots, g_t$, that is

$$x_{t+1} \in \mathrm{Span}(g_1, \ldots, g_t). \tag{3.15}$$

Let $e_1, \ldots, e_n$ be the canonical basis of $\mathbb{R}^n$, and $\mathrm{B}_2(R) = \{x \in \mathbb{R}^n : \|x\| \leq R\}$. We start with a theorem for the two non-smooth cases (convex and strongly convex).

Theorem 3.13. Let $t \leq n$, $L, R > 0$. There exists a convex and L-Lipschitz function f such that for any black-box procedure satisfying (3.15),

$$\min_{1 \leq s \leq t} f(x_s) - \min_{x \in \mathrm{B}_2(R)} f(x) \geq \frac{RL}{2(1 + \sqrt{t})}.$$

There also exists an α-strongly convex and L-lipschitz function f such that for any black-box procedure satisfying (3.15),

$$\min_{1 \leq s \leq t} f(x_s) - \min_{x \in \mathrm{B}_2\left(\frac{L}{2\alpha}\right)} f(x) \geq \frac{L^2}{8\alpha t}.$$

Note that the above result is restricted to a number of iterations smaller than the dimension, that is $t \leq n$. This restriction is of course necessary to obtain lower bounds polynomial in $1/t$: as we saw in Chapter 2 one can always obtain an exponential rate of convergence when the number of calls to the oracle is larger than the dimension.

Proof. We consider the following α-strongly convex function:

$$f(x) = \gamma \max_{1 \leq i \leq t} x(i) + \frac{\alpha}{2} \|x\|^2.$$

It is easy to see that

$$\partial f(x) = \alpha x + \gamma \mathrm{conv}\left(e_i, i : x(i) = \max_{1 \leq j \leq t} x(j)\right).$$

In particular if $\|x\| \leq R$ then for any $g \in \partial f(x)$ one has $\|g\| \leq \alpha R + \gamma$. In other words f is $(\alpha R + \gamma)$-Lipschitz on $\mathrm{B}_2(R)$.

Next we describe the first order oracle for this function: when asked for a subgradient at x, it returns $\alpha x + \gamma e_i$ where i is the *first* coordinate that satisfies $x(i) = \max_{1 \le j \le t} x(j)$. In particular when asked for a subgradient at $x_1 = 0$ it returns e_1. Thus x_2 must lie on the line generated by e_1. It is easy to see by induction that in fact x_s must lie in the linear span of $e_1, \ldots, e_{s-1}$. In particular for $s \le t$ we necessarily have $x_s(t) = 0$ and thus $f(x_s) \ge 0$.

It remains to compute the minimal value of f. Let y be such that $y(i) = -\frac{\gamma}{\alpha t}$ for $1 \le i \le t$ and $y(i) = 0$ for $t+1 \le i \le n$. It is clear that $0 \in \partial f(y)$ and thus the minimal value of f is

$$f(y) = -\frac{\gamma^2}{\alpha t} + \frac{\alpha}{2}\frac{\gamma^2}{\alpha^2 t} = -\frac{\gamma^2}{2\alpha t}.$$

Wrapping up, we proved that for any $s \le t$ one must have

$$f(x_s) - f(x^*) \ge \frac{\gamma^2}{2\alpha t}.$$

Taking $\gamma = L/2$ and $R = \frac{L}{2\alpha}$ we proved the lower bound for α-strongly convex functions (note in particular that $\|y\|^2 = \frac{\gamma^2}{\alpha^2 t} = \frac{L^2}{4\alpha^2 t} \le R^2$ with these parameters). On the other taking $\alpha = \frac{L}{R}\frac{1}{1+\sqrt{t}}$ and $\gamma = L\frac{\sqrt{t}}{1+\sqrt{t}}$ concludes the proof for convex functions (note in particular that $\|y\|^2 = \frac{\gamma^2}{\alpha^2 t} = R^2$ with these parameters). $\qquad\square$

We proceed now to the smooth case. As we will see in the following proofs we restrict our attention to quadratic functions, and it might be useful to recall that in this case one can attain the exact optimum in n calls to the oracle (see Section 2.4). We also recall that for a twice differentiable function f, β-smoothness is equivalent to the largest eigenvalue of the Hessians of f being smaller than β at any point, which we write

$$\nabla^2 f(x) \preceq \beta \mathrm{I}_n, \forall x.$$

Furthermore α-strong convexity is equivalent to

$$\nabla^2 f(x) \succeq \alpha \mathrm{I}_n, \forall x.$$

Theorem 3.14. Let $t \leq (n-1)/2$, $\beta > 0$. There exists a β-smooth convex function f such that for any black-box procedure satisfying (3.15),

$$\min_{1 \leq s \leq t} f(x_s) - f(x^*) \geq \frac{3\beta}{32} \frac{\|x_1 - x^*\|^2}{(t+1)^2}.$$

Proof. In this proof for $h : \mathbb{R}^n \to \mathbb{R}$ we denote $h^* = \inf_{x \in \mathbb{R}^n} h(x)$. For $k \leq n$ let $A_k \in \mathbb{R}^{n \times n}$ be the symmetric and tridiagonal matrix defined by

$$(A_k)_{i,j} = \begin{cases} 2, & i = j, i \leq k \\ -1, & j \in \{i-1, i+1\}, i \leq k, j \neq k+1 \\ 0, & \text{otherwise.} \end{cases}$$

It is easy to verify that $0 \preceq A_k \preceq 4I_n$ since

$$x^\top A_k x = 2 \sum_{i=1}^{k} x(i)^2 - 2 \sum_{i=1}^{k-1} x(i)x(i+1) = x(1)^2 + x(k)^2 + \sum_{i=1}^{k-1} (x(i) - x(i+1))^2.$$

We consider now the following β-smooth convex function:

$$f(x) = \frac{\beta}{8} x^\top A_{2t+1} x - \frac{\beta}{4} x^\top e_1.$$

Similarly to what happened in the proof Theorem 3.13, one can see here too that x_s must lie in the linear span of $e_1, \ldots, e_{s-1}$ (because of our assumption on the black-box procedure). In particular for $s \leq t$ we necessarily have $x_s(i) = 0$ for $i = s, \ldots, n$, which implies $x_s^\top A_{2t+1} x_s = x_s^\top A_s x_s$. In other words, if we denote

$$f_k(x) = \frac{\beta}{8} x^\top A_k x - \frac{\beta}{4} x^\top e_1,$$

then we just proved that

$$f(x_s) - f^* = f_s(x_s) - f_{2t+1}^* \geq f_s^* - f_{2t+1}^* \geq f_t^* - f_{2t+1}^*.$$

Thus it simply remains to compute the minimizer x_k^* of f_k, its norm, and the corresponding function value f_k^*.

The point x_k^* is the unique solution in the span of $e_1, \ldots, e_k$ of $A_k x = e_1$. It is easy to verify that it is defined by $x_k^*(i) = 1 - \frac{i}{k+1}$ for $i = 1, \ldots, k$. Thus we immediately have:

$$f_k^* = \frac{\beta}{8} (x_k^*)^\top A_k x_k^* - \frac{\beta}{4} (x_k^*)^\top e_1 = -\frac{\beta}{8} (x_k^*)^\top e_1 = -\frac{\beta}{8} \left(1 - \frac{1}{k+1} \right).$$

Furthermore note that

$$\|x_k^*\|^2 = \sum_{i=1}^{k} \left(1 - \frac{i}{k+1}\right)^2 = \sum_{i=1}^{k} \left(\frac{i}{k+1}\right)^2 \leq \frac{k+1}{3}.$$

Thus one obtains:

$$f_t^* - f_{2t+1}^* = \frac{\beta}{8}\left(\frac{1}{t+1} - \frac{1}{2t+2}\right) \geq \frac{3\beta}{32}\frac{\|x_{2t+1}^*\|^2}{(t+1)^2},$$

which concludes the proof. $\qquad\square$

To simplify the proof of the next theorem we will consider the limiting situation $n \to +\infty$. More precisely we assume now that we are working in $\ell_2 = \{x = (x(n))_{n\in\mathbb{N}} : \sum_{i=1}^{+\infty} x(i)^2 < +\infty\}$ rather than in $\mathbb{R}^n$. Note that all the theorems we proved in this chapter are in fact valid in an arbitrary Hilbert space $\mathcal{H}$. We chose to work in $\mathbb{R}^n$ only for clarity of the exposition.

Theorem 3.15. Let $\kappa > 1$. There exists a β-smooth and α-strongly convex function $f : \ell_2 \to \mathbb{R}$ with $\kappa = \beta/\alpha$ such that for any $t \geq 1$ and any black-box procedure satisfying (3.15) one has

$$f(x_t) - f(x^*) \geq \frac{\alpha}{2}\left(\frac{\sqrt{\kappa} - 1}{\sqrt{\kappa} + 1}\right)^{2(t-1)} \|x_1 - x^*\|^2.$$

Note that for large values of the condition number κ one has

$$\left(\frac{\sqrt{\kappa} - 1}{\sqrt{\kappa} + 1}\right)^{2(t-1)} \approx \exp\left(-\frac{4(t-1)}{\sqrt{\kappa}}\right).$$

Proof. The overall argument is similar to the proof of Theorem 3.14. Let $A : \ell_2 \to \ell_2$ be the linear operator that corresponds to the infinite tridiagonal matrix with 2 on the diagonal and -1 on the upper and lower diagonals. We consider now the following function:

$$f(x) = \frac{\alpha(\kappa - 1)}{8}\left(\langle Ax, x\rangle - 2\langle e_1, x\rangle\right) + \frac{\alpha}{2}\|x\|^2.$$

We already proved that $0 \preceq A \preceq 4I$ which easily implies that f is α-strongly convex and β-smooth. Now as always the key observation is

that for this function, thanks to our assumption on the black-box procedure, one necessarily has $x_t(i) = 0, \forall i \geq t$. This implies in particular:

$$\|x_t - x^*\|^2 \geq \sum_{i=t}^{+\infty} x^*(i)^2.$$

Furthermore since f is α-strongly convex, one has

$$f(x_t) - f(x^*) \geq \frac{\alpha}{2} \|x_t - x^*\|^2.$$

Thus it only remains to compute x^*. This can be done by differentiating f and setting the gradient to 0, which gives the following infinite set of equations

$$1 - 2\frac{\kappa + 1}{\kappa - 1}x^*(1) + x^*(2) = 0,$$

$$x^*(k-1) - 2\frac{\kappa + 1}{\kappa - 1}x^*(k) + x^*(k+1) = 0, \forall k \geq 2.$$

It is easy to verify that x^* defined by $x^*(i) = \left(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right)^i$ satisfy this infinite set of equations, and the conclusion of the theorem then follows by straightforward computations. $\qquad\square$

3.6 Geometric descent

So far our results leave a gap in the case of smooth optimization: gradient descent achieves an oracle complexity of $O(1/\varepsilon)$ (respectively $O(\kappa \log(1/\varepsilon))$ in the strongly convex case) while we proved a lower bound of $\Omega(1/\sqrt{\varepsilon})$ (respectively $\Omega(\sqrt{\kappa} \log(1/\varepsilon))$). In this section we close these gaps with the geometric descent method which was recently introduced in Bubeck et al. [2015b]. Historically the first method with optimal oracle complexity was proposed in Nemirovski and Yudin [1983]. This method, inspired by the conjugate gradient (see Section 2.4), assumes an oracle to compute *plane searches*. In Nemirovski [1982] this assumption was relaxed to a line search oracle (the geometric descent method also requires a line search oracle). Finally in Nesterov [1983] an optimal method requiring only a first order oracle was introduced. The latter algorithm, called Nesterov's accelerated gradient

descent, has been the most influential optimal method for smooth optimization up to this day. We describe and analyze this method in Section 3.7. As we shall see the intuition behind Nesterov's accelerated gradient descent (both for the derivation of the algorithm and its analysis) is not quite transparent, which motivates the present section as geometric descent has a simple geometric interpretation loosely inspired from the ellipsoid method (see Section 2.2).

We focus here on the unconstrained optimization of a smooth and strongly convex function, and we prove that geometric descent achieves the oracle complexity of $O(\sqrt{\kappa}\log(1/\varepsilon))$, thus reducing the complexity of the basic gradient descent by a factor $\sqrt{\kappa}$. We note that this improvement is quite relevant for machine learning applications. Consider for example the logistic regression problem described in Section 1.1: this is a smooth and strongly convex problem, with a smoothness of order of a numerical constant, but with strong convexity equal to the regularization parameter whose inverse can be as large as the sample size. Thus in this case κ can be of order of the sample size, and a faster rate by a factor of $\sqrt{\kappa}$ is quite significant. We also observe that this improved rate for smooth and strongly convex objectives also implies an almost optimal rate of $O(\log(1/\varepsilon)/\sqrt{\varepsilon})$ for the smooth case, as one can simply run geometric descent on the function $x \mapsto f(x) + \varepsilon\|x\|^2$.

In Section 3.6.1 we describe the basic idea of geometric descent, and we show how to obtain effortlessly a geometric method with an oracle complexity of $O(\kappa\log(1/\varepsilon))$ (i.e., similar to gradient descent). Then we explain why one should expect to be able to accelerate this method in Section 3.6.2. The geometric descent method is described precisely and analyzed in Section 3.6.3.

3.6.1 Warm-up: a geometric alternative to gradient descent

We start with some notation. Let $\mathrm{B}(x, r^2) := \{y \in \mathbb{R}^n : \|y - x\|^2 \leq r^2\}$ (note that the second argument is the radius squared), and

$$x^+ = x - \frac{1}{\beta}\nabla f(x), \text{ and } x^{++} = x - \frac{1}{\alpha}\nabla f(x).$$
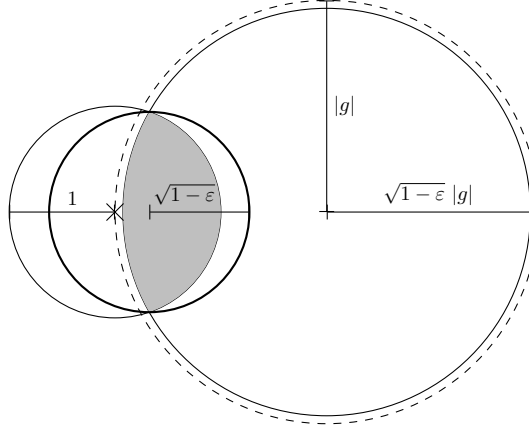
Figure 3.4: One ball shrinks.

Rewriting the definition of strong convexity (3.13) as

$$f(y) \geq f(x) + \nabla f(x)^\top (y - x) + \tfrac{\alpha}{2} \|y - x\|^2$$
$$\Leftrightarrow \ \tfrac{\alpha}{2} \|y - x + \tfrac{1}{\alpha} \nabla f(x)\|^2 \leq \tfrac{\|\nabla f(x)\|^2}{2\alpha} - (f(x) - f(y)),$$

one obtains an enclosing ball for the minimizer of f with the 0th and 1st order information at x:

$$x^* \in \mathrm{B}\left(x^{++}, \frac{\|\nabla f(x)\|^2}{\alpha^2} - \frac{2}{\alpha}(f(x) - f(x^*))\right).$$

Furthermore recall that by smoothness (see (3.5)) one has $f(x^+) \leq f(x) - \frac{1}{2\beta}\|\nabla f(x)\|^2$ which allows to *shrink* the above ball by a factor of $1 - \frac{1}{\kappa}$ and obtain the following:

$$x^* \in \mathrm{B}\left(x^{++}, \frac{\|\nabla f(x)\|^2}{\alpha^2}\left(1 - \frac{1}{\kappa}\right) - \frac{2}{\alpha}(f(x^+) - f(x^*))\right) \qquad (3.16)$$

This suggests a natural strategy: assuming that one has an enclosing ball $A := \mathrm{B}(x, R^2)$ for x^* (obtained from previous steps of the strategy), one can then enclose x^* in a ball B containing the intersection of $\mathrm{B}(x, R^2)$ and the ball $\mathrm{B}\left(x^{++}, \frac{\|\nabla f(x)\|^2}{\alpha^2}\left(1 - \frac{1}{\kappa}\right)\right)$ obtained by (3.16). Provided that the radius of B is a fraction of the radius of A, one can
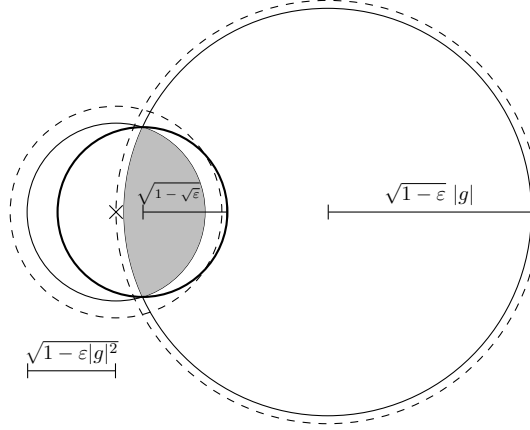
Figure 3.5: Two balls shrink.

then iterate the procedure by replacing A by B, leading to a linear convergence rate. Evaluating the rate at which the radius shrinks is an elementary calculation: for any $g \in \mathbb{R}^n$, $\varepsilon \in (0, 1)$, there exists $x \in \mathbb{R}^n$ such that

$$\mathrm{B}(0, 1) \cap \mathrm{B}(g, \|g\|^2(1 - \varepsilon)) \subset \mathrm{B}(x, 1 - \varepsilon). \qquad \text{(Figure 3.4)}$$

Thus we see that in the strategy described above, the radius squared of the enclosing ball for x^* shrinks by a factor $1 - \frac{1}{\kappa}$ at each iteration, thus matching the rate of convergence of gradient descent (see Theorem 3.10).

3.6.2 Acceleration

In the argument from the previous section we missed the following opportunity: observe that the ball $A = \mathrm{B}(x, R^2)$ was obtained by intersections of previous balls of the form given by (3.16), and thus the new value $f(x)$ could be used to reduce the radius of those previous balls too (an important caveat is that the value $f(x)$ should be smaller than the values used to build those previous balls). Potentially this could show that the optimum is in fact contained in the ball $\mathrm{B}\left(x, R^2 - \frac{1}{\kappa}\|\nabla f(x)\|^2\right)$. By taking the intersection with the ball

$\mathrm{B}\left(x^{++}, \frac{\|\nabla f(x)\|^2}{\alpha^2}\left(1 - \frac{1}{\kappa}\right)\right)$ this would allow to obtain a new ball with radius shrunk by a factor $1 - \frac{1}{\sqrt{\kappa}}$ (instead of $1 - \frac{1}{\kappa}$): indeed for any $g \in \mathbb{R}^n$, $\varepsilon \in (0,1)$, there exists $x \in \mathbb{R}^n$ such that

$$\mathrm{B}(0, 1 - \varepsilon\|g\|^2) \cap \mathrm{B}(g, \|g\|^2(1 - \varepsilon)) \subset \mathrm{B}(x, 1 - \sqrt{\varepsilon}). \qquad \text{(Figure 3.5)}$$

Thus it only remains to deal with the caveat noted above, which we do via a line search. In turns this line search might shift the new ball (3.16), and to deal with this we shall need the following strengthening of the above set inclusion (we refer to Bubeck et al. [2015b] for a simple proof of this result):

Lemma 3.16. Let $a \in \mathbb{R}^n$ and $\varepsilon \in (0,1), g \in \mathbb{R}_+$. Assume that $\|a\| \geq g$. Then there exists $c \in \mathbb{R}^n$ such that for any $\delta \geq 0$,

$$\mathrm{B}(0, 1 - \varepsilon g^2 - \delta) \cap \mathrm{B}(a, g^2(1 - \varepsilon) - \delta) \subset \mathrm{B}\left(c, 1 - \sqrt{\varepsilon} - \delta\right).$$

3.6.3   The geometric descent method

Let $x_0 \in \mathbb{R}^n$, $c_0 = x_0^{++}$, and $R_0^2 = \left(1 - \frac{1}{\kappa}\right)\frac{\|\nabla f(x_0)\|^2}{\alpha^2}$. For any $t \geq 0$ let

$$x_{t+1} = \underset{x \in \left\{(1-\lambda)c_t + \lambda x_t^+, \ \lambda \in \mathbb{R}\right\}}{\operatorname{argmin}} f(x),$$

and c_{t+1} (respectively R_{t+1}^2) be the center (respectively the squared radius) of the ball given by (the proof of) Lemma 3.16 which contains

$$\mathrm{B}\left(c_t, R_t^2 - \frac{\|\nabla f(x_{t+1})\|^2}{\alpha^2 \kappa}\right) \cap \mathrm{B}\left(x_{t+1}^{++}, \frac{\|\nabla f(x_{t+1})\|^2}{\alpha^2}\left(1 - \frac{1}{\kappa}\right)\right).$$

Formulas for c_{t+1} and R_{t+1}^2 are given at the end of this section.

Theorem 3.17. For any $t \geq 0$, one has $x^* \in \mathrm{B}(c_t, R_t^2)$, $R_{t+1}^2 \leq \left(1 - \frac{1}{\sqrt{\kappa}}\right) R_t^2$, and thus

$$\|x^* - c_t\|^2 \leq \left(1 - \frac{1}{\sqrt{\kappa}}\right)^t R_0^2.$$

Proof. We will prove a stronger claim by induction that for each $t \geq 0$, one has

$$x^* \in \mathrm{B}\left(c_t, R_t^2 - \frac{2}{\alpha}\left(f(x_t^+) - f(x^*)\right)\right).$$

The case $t = 0$ follows immediately by (3.16). Let us assume that the above display is true for some $t \geq 0$. Then using $f(x_{t+1}^+) \leq f(x_{t+1}) - \frac{1}{2\beta}\|\nabla f(x_{t+1})\|^2 \leq f(x_t^+) - \frac{1}{2\beta}\|\nabla f(x_{t+1})\|^2$, one gets

$$x^* \in \mathrm{B}\left(c_t, R_t^2 - \frac{\|\nabla f(x_{t+1})\|^2}{\alpha^2\kappa} - \frac{2}{\alpha}\left(f(x_{t+1}^+) - f(x^*)\right)\right).$$

Furthermore by (3.16) one also has

$$\mathrm{B}\left(x_{t+1}^{++}, \frac{\|\nabla f(x_{t+1})\|^2}{\alpha^2}\left(1 - \frac{1}{\kappa}\right) - \frac{2}{\alpha}\left(f(x_{t+1}^+) - f(x^*)\right)\right).$$

Thus it only remains to observe that the squared radius of the ball given by Lemma 3.16 which encloses the intersection of the two above balls is smaller than $\left(1 - \frac{1}{\sqrt{\kappa}}\right)R_t^2 - \frac{2}{\alpha}(f(x_{t+1}^+) - f(x^*))$. We apply Lemma 3.16 after moving c_t to the origin and scaling distances by R_t. We set $\varepsilon = \frac{1}{\kappa}$, $g = \frac{\|\nabla f(x_{t+1})\|}{\alpha}$, $\delta = \frac{2}{\alpha}\left(f(x_{t+1}^+) - f(x^*)\right)$ and $a = x_{t+1}^{++} - c_t$. The line search step of the algorithm implies that $\nabla f(x_{t+1})^\top (x_{t+1} - c_t) = 0$ and therefore, $\|a\| = \|x_{t+1}^{++} - c_t\| \geq \|\nabla f(x_{t+1})\|/\alpha = g$ and Lemma 3.16 applies to give the result. $\qquad\square$

One can use the following formulas for c_{t+1} and R_{t+1}^2 (they are derived from the proof of Lemma 3.16). If $|\nabla f(x_{t+1})|^2/\alpha^2 < R_t^2/2$ then one can tate $c_{t+1} = x_{t+1}^{++}$ and $R_{t+1}^2 = \frac{|\nabla f(x_{t+1})|^2}{\alpha^2}\left(1 - \frac{1}{\kappa}\right)$. On the other hand if $|\nabla f(x_{t+1})|^2/\alpha^2 \geq R_t^2/2$ then one can tate

$$c_{t+1} = c_t + \frac{R_t^2 + |x_{t+1} - c_t|^2}{2|x_{t+1}^{++} - c_t|^2}(x_{t+1}^{++} - c_t),$$

$$R_{t+1}^2 = R_t^2 - \frac{|\nabla f(x_{t+1})|^2}{\alpha^2\kappa} - \left(\frac{R_t^2 + \|x_{t+1} - c_t\|^2}{2\|x_{t+1}^{++} - c_t\|}\right)^2.$$

3.7 Nesterov's accelerated gradient descent

We describe here the original Nesterov's method which attains the optimal oracle complexity for smooth convex optimization. We give the details of the method both for the strongly convex and non-strongly convex case. We refer to Su et al. [2014] for a recent interpretation of
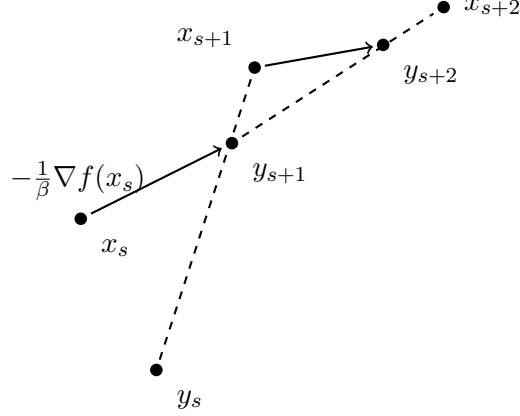
Figure 3.6: Illustration of Nesterov's accelerated gradient descent.

the method in terms of differential equations, and to Allen-Zhu and Orecchia [2014] for its relation to mirror descent (see Chapter 4).

3.7.1   The smooth and strongly convex case

Nesterov's accelerated gradient descent, illustrated in Figure 3.6, can be described as follows: Start at an arbitrary initial point $x_1 = y_1$ and then iterate the following equations for $t \geq 1$,

$$
\begin{aligned}
y_{t+1} &= x_t - \frac{1}{\beta}\nabla f(x_t), \\
x_{t+1} &= \left(1 + \frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right)y_{t+1} - \frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}y_t.
\end{aligned}
$$

Theorem 3.18. Let f be α-strongly convex and β-smooth, then Nesterov's accelerated gradient descent satisfies

$$
f(y_t) - f(x^*) \leq \frac{\alpha+\beta}{2}\|x_1 - x^*\|^2 \exp\left(-\frac{t-1}{\sqrt{\kappa}}\right).
$$

Proof. We define α-strongly convex quadratic functions $\Phi_s, s \geq 1$ by

induction as follows:

$$\Phi_1(x) = f(x_1) + \frac{\alpha}{2}\|x - x_1\|^2,$$

$$\Phi_{s+1}(x) = \left(1 - \frac{1}{\sqrt{\kappa}}\right)\Phi_s(x)$$

$$+ \frac{1}{\sqrt{\kappa}}\left(f(x_s) + \nabla f(x_s)^\top(x - x_s) + \frac{\alpha}{2}\|x - x_s\|^2\right). \qquad (3.17)$$

Intuitively Φ_s becomes a finer and finer approximation (from below) to f in the following sense:

$$\Phi_{s+1}(x) \leq f(x) + \left(1 - \frac{1}{\sqrt{\kappa}}\right)^s (\Phi_1(x) - f(x)). \qquad (3.18)$$

The above inequality can be proved immediately by induction, using the fact that by α-strong convexity one has

$$f(x_s) + \nabla f(x_s)^\top(x - x_s) + \frac{\alpha}{2}\|x - x_s\|^2 \leq f(x).$$

Equation (3.18) by itself does not say much, for it to be useful one needs to understand how "far" below f is Φ_s. The following inequality answers this question:

$$f(y_s) \leq \min_{x \in \mathbb{R}^n} \Phi_s(x). \qquad (3.19)$$

The rest of the proof is devoted to showing that (3.19) holds true, but first let us see how to combine (3.18) and (3.19) to obtain the rate given by the theorem (we use that by β-smoothness one has $f(x) - f(x^*) \leq \frac{\beta}{2}\|x - x^*\|^2$):

$$\begin{aligned}
f(y_t) - f(x^*) &\leq \Phi_t(x^*) - f(x^*) \\
&\leq \left(1 - \frac{1}{\sqrt{\kappa}}\right)^{t-1}(\Phi_1(x^*) - f(x^*)) \\
&\leq \frac{\alpha + \beta}{2}\|x_1 - x^*\|^2\left(1 - \frac{1}{\sqrt{\kappa}}\right)^{t-1}.
\end{aligned}$$

We now prove (3.19) by induction (note that it is true at $s = 1$ since $x_1 = y_1$). Let $\Phi_s^* = \min_{x \in \mathbb{R}^n} \Phi_s(x)$. Using the definition of y_{s+1} (and

β-smoothness), convexity, and the induction hypothesis, one gets

$$
\begin{aligned}
f(y_{s+1}) \ &\leq\ f(x_s) - \frac{1}{2\beta}\|\nabla f(x_s)\|^2 \\
&=\ \left(1 - \frac{1}{\sqrt{\kappa}}\right) f(y_s) + \left(1 - \frac{1}{\sqrt{\kappa}}\right)(f(x_s) - f(y_s)) \\
&\quad + \frac{1}{\sqrt{\kappa}} f(x_s) - \frac{1}{2\beta}\|\nabla f(x_s)\|^2 \\
&\leq\ \left(1 - \frac{1}{\sqrt{\kappa}}\right)\Phi_s^* + \left(1 - \frac{1}{\sqrt{\kappa}}\right)\nabla f(x_s)^\top (x_s - y_s) \\
&\quad + \frac{1}{\sqrt{\kappa}} f(x_s) - \frac{1}{2\beta}\|\nabla f(x_s)\|^2.
\end{aligned}
$$

Thus we now have to show that

$$
\begin{aligned}
\Phi_{s+1}^* \ &\geq\ \left(1 - \frac{1}{\sqrt{\kappa}}\right)\Phi_s^* + \left(1 - \frac{1}{\sqrt{\kappa}}\right)\nabla f(x_s)^\top (x_s - y_s) \\
&\quad + \frac{1}{\sqrt{\kappa}} f(x_s) - \frac{1}{2\beta}\|\nabla f(x_s)\|^2. \tag{3.20}
\end{aligned}
$$

To prove this inequality we have to understand better the functions Φ_s. First note that $\nabla^2\Phi_s(x) = \alpha I_n$ (immediate by induction) and thus Φ_s has to be of the following form:

$$
\Phi_s(x) = \Phi_s^* + \frac{\alpha}{2}\|x - v_s\|^2,
$$

for some $v_s \in \mathbb{R}^n$. Now observe that by differentiating (3.17) and using the above form of Φ_s one obtains

$$
\nabla\Phi_{s+1}(x) = \alpha\left(1 - \frac{1}{\sqrt{\kappa}}\right)(x - v_s) + \frac{1}{\sqrt{\kappa}}\nabla f(x_s) + \frac{\alpha}{\sqrt{\kappa}}(x - x_s).
$$

In particular Φ_{s+1} is by definition minimized at v_{s+1} which can now be defined by induction using the above identity, precisely:

$$
v_{s+1} = \left(1 - \frac{1}{\sqrt{\kappa}}\right)v_s + \frac{1}{\sqrt{\kappa}}x_s - \frac{1}{\alpha\sqrt{\kappa}}\nabla f(x_s). \tag{3.21}
$$

Using the form of Φ_s and Φ_{s+1}, as well as the original definition (3.17) one gets the following identity by evaluating Φ_{s+1} at x_s:

$$
\begin{aligned}
\Phi_{s+1}^* &+ \frac{\alpha}{2}\|x_s - v_{s+1}\|^2 \\
&= \left(1 - \frac{1}{\sqrt{\kappa}}\right)\Phi_s^* + \frac{\alpha}{2}\left(1 - \frac{1}{\sqrt{\kappa}}\right)\|x_s - v_s\|^2 + \frac{1}{\sqrt{\kappa}}f(x_s). \tag{3.22}
\end{aligned}
$$

Note that thanks to (3.21) one has

$$
\begin{aligned}
\|x_s - v_{s+1}\|^2 &= \left(1 - \frac{1}{\sqrt{\kappa}}\right)^2 \|x_s - v_s\|^2 + \frac{1}{\alpha^2 \kappa}\|\nabla f(x_s)\|^2 \\
&\quad - \frac{2}{\alpha\sqrt{\kappa}}\left(1 - \frac{1}{\sqrt{\kappa}}\right)\nabla f(x_s)^\top (v_s - x_s),
\end{aligned}
$$

which combined with (3.22) yields

$$
\begin{aligned}
\Phi_{s+1}^* &= \left(1 - \frac{1}{\sqrt{\kappa}}\right)\Phi_s^* + \frac{1}{\sqrt{\kappa}}f(x_s) + \frac{\alpha}{2\sqrt{\kappa}}\left(1 - \frac{1}{\sqrt{\kappa}}\right)\|x_s - v_s\|^2 \\
&\quad - \frac{1}{2\beta}\|\nabla f(x_s)\|^2 + \frac{1}{\sqrt{\kappa}}\left(1 - \frac{1}{\sqrt{\kappa}}\right)\nabla f(x_s)^\top (v_s - x_s).
\end{aligned}
$$

Finally we show by induction that $v_s - x_s = \sqrt{\kappa}(x_s - y_s)$, which concludes the proof of (3.20) and thus also concludes the proof of the theorem:

$$
\begin{aligned}
v_{s+1} - x_{s+1} &= \left(1 - \frac{1}{\sqrt{\kappa}}\right)v_s + \frac{1}{\sqrt{\kappa}}x_s - \frac{1}{\alpha\sqrt{\kappa}}\nabla f(x_s) - x_{s+1} \\
&= \sqrt{\kappa}x_s - (\sqrt{\kappa} - 1)y_s - \frac{\sqrt{\kappa}}{\beta}\nabla f(x_s) - x_{s+1} \\
&= \sqrt{\kappa}y_{s+1} - (\sqrt{\kappa} - 1)y_s - x_{s+1} \\
&= \sqrt{\kappa}(x_{s+1} - y_{s+1}),
\end{aligned}
$$

where the first equality comes from (3.21), the second from the induction hypothesis, the third from the definition of y_{s+1} and the last one from the definition of x_{s+1}. □

3.7.2 The smooth case

In this section we show how to adapt Nesterov's accelerated gradient descent for the case $\alpha = 0$, using a time-varying combination of the elements in the primary sequence (y_t). First we define the following sequences:

$$
\lambda_0 = 0, \ \lambda_t = \frac{1 + \sqrt{1 + 4\lambda_{t-1}^2}}{2}, \ \text{and} \ \gamma_t = \frac{1 - \lambda_t}{\lambda_{t+1}}.
$$

(Note that $\gamma_t \leq 0$.) Now the algorithm is simply defined by the following equations, with $x_1 = y_1$ an arbitrary initial point,

$$
\begin{aligned}
y_{t+1} &= x_t - \frac{1}{\beta}\nabla f(x_t), \\
x_{t+1} &= (1 - \gamma_s)y_{t+1} + \gamma_t y_t.
\end{aligned}
$$

Theorem 3.19. Let f be a convex and β-smooth function, then Nesterov's accelerated gradient descent satisfies

$$
f(y_t) - f(x^*) \leq \frac{2\beta\|x_1 - x^*\|^2}{t^2}.
$$

We follow here the proof of Beck and Teboulle [2009]. We also refer to Tseng [2008] for a proof with simpler step-sizes.

Proof. Using the unconstrained version of Lemma 3.6 one obtains

$$
\begin{aligned}
&f(y_{s+1}) - f(y_s) \\
&\leq \nabla f(x_s)^\top (x_s - y_s) - \frac{1}{2\beta}\|\nabla f(x_s)\|^2 \\
&= \beta(x_s - y_{s+1})^\top (x_s - y_s) - \frac{\beta}{2}\|x_s - y_{s+1}\|^2. \qquad (3.23)
\end{aligned}
$$

Similarly we also get

$$
f(y_{s+1}) - f(x^*) \leq \beta(x_s - y_{s+1})^\top (x_s - x^*) - \frac{\beta}{2}\|x_s - y_{s+1}\|^2. \quad (3.24)
$$

Now multiplying (3.23) by $(\lambda_s - 1)$ and adding the result to (3.24), one obtains with $\delta_s = f(y_s) - f(x^*)$,

$$
\begin{aligned}
&\lambda_s \delta_{s+1} - (\lambda_s - 1)\delta_s \\
&\leq \beta(x_s - y_{s+1})^\top (\lambda_s x_s - (\lambda_s - 1)y_s - x^*) - \frac{\beta}{2}\lambda_s\|x_s - y_{s+1}\|^2.
\end{aligned}
$$

Multiplying this inequality by λ_s and using that by definition $\lambda_{s-1}^2 = \lambda_s^2 - \lambda_s$, as well as the elementary identity $2a^\top b - \|a\|^2 = \|b\|^2 - \|b - a\|^2$,

one obtains

$$\lambda_s^2 \delta_{s+1} - \lambda_{s-1}^2 \delta_s$$
$$\leq \frac{\beta}{2} \left(2\lambda_s (x_s - y_{s+1})^\top (\lambda_s x_s - (\lambda_s - 1)y_s - x^*) - \|\lambda_s (y_{s+1} - x_s)\|^2 \right)$$
$$= \frac{\beta}{2} \left(\|\lambda_s x_s - (\lambda_s - 1)y_s - x^*\|^2 - \|\lambda_s y_{s+1} - (\lambda_s - 1)y_s - x^*\|^2 \right).$$
$$(3.25)$$

Next remark that, by definition, one has

$$x_{s+1} = y_{s+1} + \gamma_s(y_s - y_{s+1})$$
$$\Leftrightarrow \lambda_{s+1} x_{s+1} = \lambda_{s+1} y_{s+1} + (1 - \lambda_s)(y_s - y_{s+1})$$
$$\Leftrightarrow \lambda_{s+1} x_{s+1} - (\lambda_{s+1} - 1)y_{s+1} = \lambda_s y_{s+1} - (\lambda_s - 1)y_s. \qquad (3.26)$$

Putting together (3.25) and (3.26) one gets with $u_s = \lambda_s x_s - (\lambda_s - 1)y_s - x^*$,

$$\lambda_s^2 \delta_{s+1} - \lambda_{s-1}^2 \delta_s^2 \leq \frac{\beta}{2} \left(\|u_s\|^2 - \|u_{s+1}\|^2 \right).$$

Summing these inequalities from $s = 1$ to $s = t - 1$ one obtains:

$$\delta_t \leq \frac{\beta}{2\lambda_{t-1}^2} \|u_1\|^2.$$

By induction it is easy to see that $\lambda_{t-1} \geq \frac{t}{2}$ which concludes the proof.

$\square$

4

Almost dimension-free convex optimization in non-Euclidean spaces

---

In the previous chapter we showed that dimension-free oracle complexity is possible when the objective function f and the constraint set $\mathcal{X}$ are well-behaved in the Euclidean norm; e.g. if for all points $x \in \mathcal{X}$ and all subgradients $g \in \partial f(x)$, one has that $\|x\|_2$ and $\|g\|_2$ are independent of the ambient dimension n. If this assumption is not met then the gradient descent techniques of Chapter 3 may lose their dimension-free convergence rates. For instance consider a differentiable convex function f defined on the Euclidean ball $\mathrm{B}_{2,n}$ and such that $\|\nabla f(x)\|_\infty \leq 1, \forall x \in \mathrm{B}_{2,n}$. This implies that $\|\nabla f(x)\|_2 \leq \sqrt{n}$, and thus projected gradient descent will converge to the minimum of f on $\mathrm{B}_{2,n}$ at a rate $\sqrt{n/t}$. In this chapter we describe the method of Nemirovski and Yudin [1983], known as mirror descent, which allows to find the minimum of such functions f over the ℓ_1-ball (instead of the Euclidean ball) at the much faster rate $\sqrt{\log(n)/t}$. This is only one example of the potential of mirror descent. This chapter is devoted to the description of mirror descent and some of its alternatives. The presentation is inspired from Beck and Teboulle [2003], [Chapter 11, Cesa-Bianchi and Lugosi [2006]], Rakhlin [2009], Hazan [2011], Bubeck [2011].

In order to describe the intuition behind the method let us abstract the situation for a moment and forget that we are doing optimization in finite dimension. We already observed that projected gradient descent works in an arbitrary Hilbert space $\mathcal{H}$. Suppose now that we are interested in the more general situation of optimization in some Banach space $\mathcal{B}$. In other words the norm that we use to measure the various quantity of interest does not derive from an inner product (think of $\mathcal{B} = \ell_1$ for example). In that case the gradient descent strategy does not even make sense: indeed the gradients (more formally the Fréchet derivative) $\nabla f(x)$ are elements of the dual space $\mathcal{B}^*$ and thus one cannot perform the computation $x - \eta \nabla f(x)$ (it simply does not make sense). We did not have this problem for optimization in a Hilbert space $\mathcal{H}$ since by Riesz representation theorem $\mathcal{H}^*$ is isometric to $\mathcal{H}$. The great insight of Nemirovski and Yudin is that one can still do a gradient descent by first mapping the point $x \in \mathcal{B}$ into the dual space $\mathcal{B}^*$, then performing the gradient update in the dual space, and finally mapping back the resulting point to the primal space $\mathcal{B}$. Of course the new point in the primal space might lie outside of the constraint set $\mathcal{X} \subset \mathcal{B}$ and thus we need a way to project back the point on the constraint set $\mathcal{X}$. Both the primal/dual mapping and the projection are based on the concept of a *mirror map* which is the key element of the scheme. Mirror maps are defined in Section 4.1, and the above scheme is formally described in Section 4.2.

In the rest of this chapter we fix an arbitrary norm $\|\cdot\|$ on $\mathbb{R}^n$, and a compact convex set $\mathcal{X} \subset \mathbb{R}^n$. The dual norm $\|\cdot\|_*$ is defined as $\|g\|_* = \sup_{x \in \mathbb{R}^n : \|x\| \leq 1} g^\top x$. We say that a convex function $f : \mathcal{X} \to \mathbb{R}$ is (i) L-Lipschitz w.r.t. $\|\cdot\|$ if $\forall x \in \mathcal{X}, g \in \partial f(x), \|g\|_* \leq L$, (ii) β-smooth w.r.t. $\|\cdot\|$ if $\|\nabla f(x) - \nabla f(y)\|_* \leq \beta \|x - y\|, \forall x, y \in \mathcal{X}$, and (iii) α-strongly convex w.r.t. $\|\cdot\|$ if

$$f(x) - f(y) \leq g^\top (x - y) - \frac{\alpha}{2} \|x - y\|^2, \forall x, y \in \mathcal{X}, g \in \partial f(x).$$

We also define the Bregman divergence associated to f as

$$D_f(x, y) = f(x) - f(y) - \nabla f(y)^\top (x - y).$$

The following identity will be useful several times:

$$(\nabla f(x) - \nabla f(y))^\top (x - z) = D_f(x, y) + D_f(z, x) - D_f(z, y). \quad (4.1)$$

4.1   Mirror maps

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 $\mathcal{X} \cap \mathcal{D} \neq \emptyset$. We say that $\Phi : \mathcal{D} \to \mathbb{R}$ is a mirror map if it safisfies the following properties[1]:

(i) Φ is strictly convex and differentiable.

(ii) The gradient of Φ takes all possible values, that is $\nabla\Phi(\mathcal{D}) = \mathbb{R}^n$.

(iii) The gradient of Φ diverges on the boundary of $\mathcal{D}$, that is

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

In mirror descent the gradient of the mirror map Φ 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 $\nabla\Phi(x)$, from which one takes a gradient step to get to $\nabla\Phi(x) - \eta\nabla f(x)$. Property (ii) then allows us to write the resulting point as $\nabla\Phi(y) = \nabla\Phi(x) - \eta\nabla f(x)$ 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 Φ. Precisely one defines

$$\Pi_{\mathcal{X}}^{\Phi}(y) = \underset{x \in \mathcal{X} \cap \mathcal{D}}{\arg\min} \, D_{\Phi}(x, y).$$

Property (i) and (iii) ensures the existence and uniqueness of this projection (in particular since $x \mapsto D_{\Phi}(x, y)$ is locally increasing on the boundary of $\mathcal{D}$). The following lemma shows that the Bregman divergence essentially behaves as the Euclidean norm squared in terms of projections (recall Lemma 3.1).

Lemma 4.1. Let $x \in \mathcal{X} \cap \mathcal{D}$ and $y \in \mathcal{D}$, then

$$(\nabla\Phi(\Pi_{\mathcal{X}}^{\Phi}(y)) - \nabla\Phi(y))^{\top}(\Pi_{\mathcal{X}}^{\Phi}(y) - x) \leq 0,$$

---

[1]Assumption (ii) can be relaxed in some cases, see for example Audibert et al. [2014].
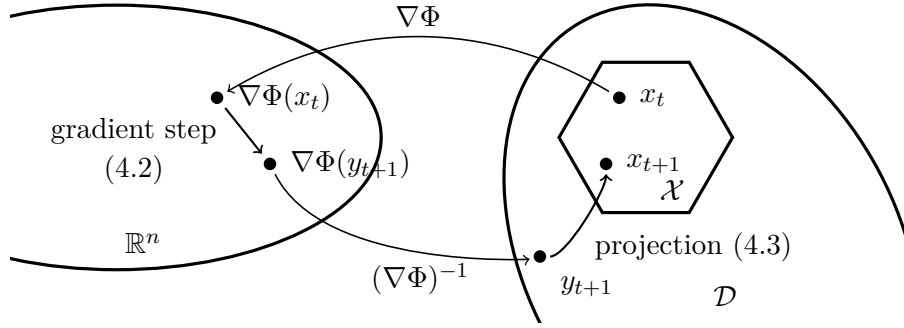
Figure 4.1: Illustration of mirror descent.

which also implies

$$D_\Phi(x, \Pi_\mathcal{X}^\Phi(y)) + D_\Phi(\Pi_\mathcal{X}^\Phi(y), y) \le D_\Phi(x, y).$$

Proof. The proof is an immediate corollary of Proposition 1.3 together with the fact that $\nabla_x D_\Phi(x, y) = \nabla\Phi(x) - \nabla\Phi(y)$. $\qquad\square$

4.2 Mirror descent

We can now describe the mirror descent strategy based on a mirror map Φ. Let $x_1 \in \mathrm{argmin}_{x \in \mathcal{X} \cap \mathcal{D}} \Phi(x)$. Then for $t \ge 1$, let $y_{t+1} \in \mathcal{D}$ such that

$$\nabla\Phi(y_{t+1}) = \nabla\Phi(x_t) - \eta g_t, \text{ where } g_t \in \partial f(x_t), \tag{4.2}$$

and

$$x_{t+1} \in \Pi_\mathcal{X}^\Phi(y_{t+1}). \tag{4.3}$$

See Figure 4.1 for an illustration of this procedure.

Theorem 4.2. Let Φ be a mirror map ρ-strongly convex on $\mathcal{X} \cap \mathcal{D}$ w.r.t. $\|\cdot\|$. Let $R^2 = \sup_{x \in \mathcal{X} \cap \mathcal{D}} \Phi(x) - \Phi(x_1)$, and f be convex and

L-Lipschitz w.r.t. $\| \cdot \|$. Then mirror descent with $\eta = \frac{R}{L}\sqrt{\frac{2\rho}{t}}$ satisfies

$$f\left(\frac{1}{t}\sum_{s=1}^{t} x_s\right) - f(x^*) \leq RL\sqrt{\frac{2}{\rho t}}.$$

Proof. Let $x \in \mathcal{X} \cap \mathcal{D}$. The claimed bound will be obtained by taking a limit $x \to x^*$. Now by convexity of f, the definition of mirror descent, equation (4.1), and Lemma 4.1, one has

$$f(x_s) - f(x)$$
$$\leq g_s^\top (x_s - x)$$
$$= \frac{1}{\eta}(\nabla\Phi(x_s) - \nabla\Phi(y_{s+1}))^\top (x_s - x)$$
$$= \frac{1}{\eta}\Big(D_\Phi(x, x_s) + D_\Phi(x_s, y_{s+1}) - D_\Phi(x, y_{s+1})\Big)$$
$$\leq \frac{1}{\eta}\Big(D_\Phi(x, x_s) + D_\Phi(x_s, y_{s+1}) - D_\Phi(x, x_{s+1}) - D_\Phi(x_{s+1}, y_{s+1})\Big).$$

The term $D_\Phi(x, x_s) - D_\Phi(x, x_{s+1})$ will lead to a telescopic sum when summing over $s = 1$ to $s = t$, and it remains to bound the other term as follows using ρ-strong convexity of the mirror map and $az - bz^2 \leq \frac{a^2}{4b}, \forall z \in \mathbb{R}$:

$$D_\Phi(x_s, y_{s+1}) - D_\Phi(x_{s+1}, y_{s+1})$$
$$= \Phi(x_s) - \Phi(x_{s+1}) - \nabla\Phi(y_{s+1})^\top (x_s - x_{s+1})$$
$$\leq (\nabla\Phi(x_s) - \nabla\Phi(y_{s+1}))^\top (x_s - x_{s+1}) - \frac{\rho}{2}\|x_s - x_{s+1}\|^2$$
$$= \eta g_s^\top (x_s - x_{s+1}) - \frac{\rho}{2}\|x_s - x_{s+1}\|^2$$
$$\leq \eta L\|x_s - x_{s+1}\| - \frac{\rho}{2}\|x_s - x_{s+1}\|^2$$
$$\leq \frac{(\eta L)^2}{2\rho}.$$

We proved

$$\sum_{s=1}^{t}\Big(f(x_s) - f(x)\Big) \leq \frac{D_\Phi(x, x_1)}{\eta} + \eta\frac{L^2 t}{2\rho},$$

which concludes the proof up to trivial computation. $\qquad\square$

We observe that one can rewrite mirror descent as follows:

$$
\begin{aligned}
x_{t+1} &= \operatorname*{argmin}_{x \in \mathcal{X} \cap \mathcal{D}} \; D_\Phi(x, y_{t+1}) \\
&= \operatorname*{argmin}_{x \in \mathcal{X} \cap \mathcal{D}} \; \Phi(x) - \nabla\Phi(y_{t+1})^\top x \qquad (4.4) \\
&= \operatorname*{argmin}_{x \in \mathcal{X} \cap \mathcal{D}} \; \Phi(x) - (\nabla\Phi(x_t) - \eta g_t)^\top x \\
&= \operatorname*{argmin}_{x \in \mathcal{X} \cap \mathcal{D}} \; \eta g_t^\top x + D_\Phi(x, x_t). \qquad (4.5)
\end{aligned}
$$

This last expression is often taken as the definition of mirror descent (see Beck and Teboulle [2003]). It gives a proximal point of view on mirror descent: the method is trying to minimize the local linearization of the function while not moving too far away from the previous point, with distances measured via the Bregman divergence of the mirror map.

4.3 Standard setups for mirror descent

"Ball setup". The simplest version of mirror descent is obtained by taking $\Phi(x) = \frac{1}{2}\|x\|_2^2$ on $\mathcal{D} = \mathbb{R}^n$. The function Φ is a mirror map strongly convex w.r.t. $\|\cdot\|_2$, and furthermore the associated Bregman divergence is given by $D_\Phi(x, y) = \frac{1}{2}\|x - y\|_2^2$. Thus in that case mirror descent is exactly equivalent to projected subgradient descent, and the rate of convergence obtained in Theorem 4.2 recovers our earlier result on projected subgradient descent.

"Simplex setup". A more interesting choice of a mirror map is given by the negative entropy

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

on $\mathcal{D} = \mathbb{R}^n_{++}$. In that case the gradient update $\nabla\Phi(y_{t+1}) = \nabla\Phi(x_t) - \eta\nabla f(x_t)$ can be written equivalently as

$$
y_{t+1}(i) = x_t(i) \exp\big(-\eta[\nabla f(x_t)](i)\big), \; i = 1, \ldots, n.
$$

The Bregman divergence of this mirror map is given by $D_\Phi(x, y) = \sum_{i=1}^{n} x(i) \log \frac{x(i)}{y(i)}$ (also known as the Kullback-Leibler divergence). It

is easy to verify that the projection with respect to this Bregman divergence on the simplex $\Delta_n = \{x \in \mathbb{R}^n_+ : \sum_{i=1}^n x(i) = 1\}$ amounts to a simple renormalization $y \mapsto y/\|y\|_1$. Furthermore it is also easy to verify that Φ is 1-strongly convex w.r.t. $\|\cdot\|_1$ on Δ_n (this result is known as Pinsker's inequality). Note also that for $\mathcal{X} = \Delta_n$ one has $x_1 = (1/n, \ldots, 1/n)$ and $R^2 = \log n$.

The above observations imply that when minimizing on the simplex Δ_n a function f with subgradients bounded in ℓ_∞-norm, mirror descent with the negative entropy achieves a rate of convergence of order $\sqrt{\frac{\log n}{t}}$. On the other hand the regular subgradient descent achieves only a rate of order $\sqrt{\frac{n}{t}}$ in this case!

"Spectrahedron setup". We consider here functions defined on matrices, and we are interested in minimizing a function f on the *spectrahedron* $\mathcal{S}_n$ defined as:

$$\mathcal{S}_n = \{X \in \mathbb{S}^n_+ : \text{Tr}(X) = 1\}.$$

In this setting we consider the mirror map on $\mathcal{D} = \mathbb{S}^n_{++}$ given by the negative von Neumann entropy:

$$\Phi(X) = \sum_{i=1}^n \lambda_i(X) \log \lambda_i(X),$$

where $\lambda_1(X), \ldots, \lambda_n(X)$ are the eigenvalues of X. It can be shown that the gradient update $\nabla \Phi(Y_{t+1}) = \nabla \Phi(X_t) - \eta \nabla f(X_t)$ can be written equivalently as

$$Y_{t+1} = \exp\left(\log X_t - \eta \nabla f(X_t) \right),$$

where the matrix exponential and matrix logarithm are defined as usual. Furthermore the projection on $\mathcal{S}_n$ is a simple trace renormalization.

With highly non-trivial computation one can show that Φ is $\frac{1}{2}$-strongly convex with respect to the Schatten 1-norm defined as

$$\|X\|_1 = \sum_{i=1}^n \lambda_i(X).$$

It is easy to see that for $\mathcal{X} = \mathcal{S}_n$ one has $x_1 = \frac{1}{n}\mathrm{I}_n$ and $R^2 = \log n$. In other words the rate of convergence for optimization on the spectrahedron is the same than on the simplex!

4.4 Lazy mirror descent, aka Nesterov's dual averaging

In this section we consider a slightly more efficient version of mirror descent for which we can prove that Theorem 4.2 still holds true. This alternative algorithm can be advantageous in some situations (such as distributed settings), but the basic mirror descent scheme remains important for extensions considered later in this text (saddle points, stochastic oracles, ...).

In lazy mirror descent, also commonly known as Nesterov's dual averaging or simply dual averaging, one replaces (4.2) by

$$\nabla\Phi(y_{t+1}) = \nabla\Phi(y_t) - \eta g_t,$$

and also y_1 is such that $\nabla\Phi(y_1) = 0$. In other words instead of going back and forth between the primal and the dual, dual averaging simply averages the gradients in the dual, and if asked for a point in the primal it simply maps the current dual point to the primal using the same methodology as mirror descent. In particular using (4.4) one immediately sees that dual averaging is defined by:

$$x_t = \operatorname*{argmin}_{x \in \mathcal{X} \cap \mathcal{D}} \ \eta \sum_{s=1}^{t-1} g_s^\top x + \Phi(x). \tag{4.6}$$

Theorem 4.3. Let Φ be a mirror map ρ-strongly convex on $\mathcal{X} \cap \mathcal{D}$ w.r.t. $\|\cdot\|$. Let $R^2 = \sup_{x \in \mathcal{X} \cap \mathcal{D}} \Phi(x) - \Phi(x_1)$, and f be convex and L-Lipschitz w.r.t. $\|\cdot\|$. Then dual averaging with $\eta = \frac{R}{L}\sqrt{\frac{\rho}{2t}}$ satisfies

$$f\left(\frac{1}{t}\sum_{s=1}^{t} x_s\right) - f(x^*) \le 2RL\sqrt{\frac{2}{\rho t}}.$$

Proof. We define $\psi_t(x) = \eta \sum_{s=1}^{t} g_s^\top x + \Phi(x)$, so that $x_t \in \operatorname{argmin}_{x \in \mathcal{X} \cap \mathcal{D}} \psi_{t-1}(x)$. Since Φ is ρ-strongly convex one clearly has that

ψ_t is ρ-strongly convex, and thus

$$
\begin{aligned}
\psi_t(x_{t+1}) - \psi_t(x_t) &\leq \nabla\psi_t(x_{t+1})^\top(x_{t+1} - x_t) - \frac{\rho}{2}\|x_{t+1} - x_t\|^2 \\
&\leq -\frac{\rho}{2}\|x_{t+1} - x_t\|^2,
\end{aligned}
$$

where the second inequality comes from the first order optimality condition for x_{t+1} (see Proposition 1.3). Next observe that

$$
\begin{aligned}
\psi_t(x_{t+1}) - \psi_t(x_t) &= \psi_{t-1}(x_{t+1}) - \psi_{t-1}(x_t) + \eta g_t^\top(x_{t+1} - x_t) \\
&\geq \eta g_t^\top(x_{t+1} - x_t).
\end{aligned}
$$

Putting together the two above displays and using Cauchy-Schwarz (with the assumption $\|g_t\|_* \leq L$) one obtains

$$
\frac{\rho}{2}\|x_{t+1} - x_t\|^2 \leq \eta g_t^\top(x_t - x_{t+1}) \leq \eta L\|x_t - x_{t+1}\|.
$$

In particular this shows that $\|x_{t+1} - x_t\| \leq \frac{2\eta L}{\rho}$ and thus with the above display

$$
g_t^\top(x_t - x_{t+1}) \leq \frac{2\eta L^2}{\rho}. \tag{4.7}
$$

Now we claim that for any $x \in \mathcal{X} \cap \mathcal{D}$,

$$
\sum_{s=1}^{t} g_s^\top(x_s - x) \leq \sum_{s=1}^{t} g_s^\top(x_s - x_{s+1}) + \frac{\Phi(x) - \Phi(x_1)}{\eta}, \tag{4.8}
$$

which would clearly conclude the proof thanks to (4.7) and straightforward computations. Equation (4.8) is equivalent to

$$
\sum_{s=1}^{t} g_s^\top x_{s+1} + \frac{\Phi(x_1)}{\eta} \leq \sum_{s=1}^{t} g_s^\top x + \frac{\Phi(x)}{\eta},
$$

and we now prove the latter equation by induction. At $t = 0$ it is true since $x_1 \in \operatorname{argmin}_{x \in \mathcal{X} \cap \mathcal{D}} \Phi(x)$. The following inequalities prove the inductive step, where we use the induction hypothesis at $x = x_{t+1}$ for the first inequality, and the definition of x_{t+1} for the second inequality:

$$
\sum_{s=1}^{t} g_s^\top x_{s+1} + \frac{\Phi(x_1)}{\eta} \leq g_t^\top x_{t+1} + \sum_{s=1}^{t-1} g_s^\top x_{t+1} + \frac{\Phi(x_{t+1})}{\eta} \leq \sum_{s=1}^{t} g_s^\top x + \frac{\Phi(x)}{\eta}.
$$

$\square$

4.5 Mirror prox

It can be shown that mirror descent accelerates for smooth functions to the rate $1/t$. We will prove this result in Chapter 6 (see Theorem 6.3). We describe here a variant of mirror descent which also attains the rate $1/t$ for smooth functions. This method is called mirror prox and it was introduced in Nemirovski [2004a]. The true power of mirror prox will reveal itself later in the text when we deal with smooth representations of non-smooth functions as well as stochastic oracles[2].

Mirror prox is described by the following equations:

$$\nabla \Phi(y'_{t+1}) = \nabla \Phi(x_t) - \eta \nabla f(x_t),$$

$$y_{t+1} \in \operatorname*{argmin}_{x \in \mathcal{X} \cap \mathcal{D}} D_\Phi(x, y'_{t+1}),$$

$$\nabla \Phi(x'_{t+1}) = \nabla \Phi(x_t) - \eta \nabla f(y_{t+1}),$$

$$x_{t+1} \in \operatorname*{argmin}_{x \in \mathcal{X} \cap \mathcal{D}} D_\Phi(x, x'_{t+1}).$$

In words the algorithm first makes a step of mirror descent to go from x_t to y_{t+1}, and then it makes a similar step to obtain x_{t+1}, starting again from x_t but this time using the gradient of f evaluated at y_{t+1} (instead of x_t), see Figure 4.2 for an illustration. The following result justifies the procedure.

Theorem 4.4. Let Φ be a mirror map ρ-strongly convex on $\mathcal{X} \cap \mathcal{D}$ w.r.t. $\|\cdot\|$. Let $R^2 = \sup_{x \in \mathcal{X} \cap \mathcal{D}} \Phi(x) - \Phi(x_1)$, and f be convex and β-smooth w.r.t. $\|\cdot\|$. Then mirror prox with $\eta = \frac{\rho}{\beta}$ satisfies

$$f\left(\frac{1}{t} \sum_{s=1}^{t} y_{s+1}\right) - f(x^*) \leq \frac{\beta R^2}{\rho t}.$$

---

[2]Basically mirror prox allows for a smooth vector field point of view (see Section 4.6), while mirror descent does not.
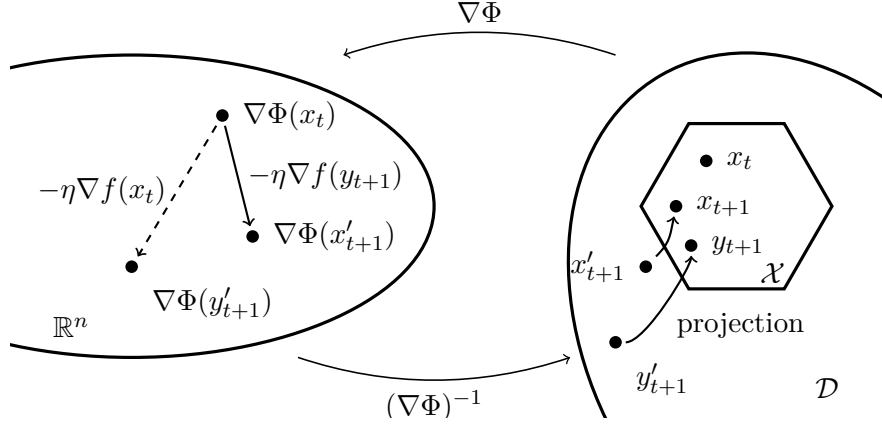
Figure 4.2: Illustration of mirror prox.

Proof. Let $x \in \mathcal{X} \cap \mathcal{D}$. We write

$$
\begin{aligned}
f(y_{t+1}) - f(x) &\leq \nabla f(y_{t+1})^\top (y_{t+1} - x) \\
&= \nabla f(y_{t+1})^\top (x_{t+1} - x) + \nabla f(x_t)^\top (y_{t+1} - x_{t+1}) \\
&\quad + (\nabla f(y_{t+1}) - \nabla f(x_t))^\top (y_{t+1} - x_{t+1}).
\end{aligned}
$$

We will now bound separately these three terms. For the first one, using the definition of the method, Lemma 4.1, and equation (4.1), one gets

$$
\begin{aligned}
&\eta \nabla f(y_{t+1})^\top (x_{t+1} - x) \\
&= (\nabla \Phi(x_t) - \nabla \Phi(x'_{t+1}))^\top (x_{t+1} - x) \\
&\leq (\nabla \Phi(x_t) - \nabla \Phi(x_{t+1}))^\top (x_{t+1} - x) \\
&= D_\Phi(x, x_t) - D_\Phi(x, x_{t+1}) - D_\Phi(x_{t+1}, x_t).
\end{aligned}
$$

For the second term using the same properties than above and the

strong-convexity of the mirror map one obtains

$$
\begin{aligned}
&\eta \nabla f(x_t)^\top (y_{t+1} - x_{t+1}) \\
&= (\nabla \Phi(x_t) - \nabla \Phi(y'_{t+1}))^\top (y_{t+1} - x_{t+1}) \\
&\leq (\nabla \Phi(x_t) - \nabla \Phi(y_{t+1}))^\top (y_{t+1} - x_{t+1}) \\
&= D_\Phi(x_{t+1}, x_t) - D_\Phi(x_{t+1}, y_{t+1}) - D_\Phi(y_{t+1}, x_t) \qquad (4.9) \\
&\leq D_\Phi(x_{t+1}, x_t) - \frac{\rho}{2}\|x_{t+1} - y_{t+1}\|^2 - \frac{\rho}{2}\|y_{t+1} - x_t\|^2.
\end{aligned}
$$

Finally for the last term, using Cauchy-Schwarz, β-smoothness, and $2ab \leq a^2 + b^2$ one gets

$$
\begin{aligned}
&(\nabla f(y_{t+1}) - \nabla f(x_t))^\top (y_{t+1} - x_{t+1}) \\
&\leq \|\nabla f(y_{t+1}) - \nabla f(x_t)\|_* \cdot \|y_{t+1} - x_{t+1}\| \\
&\leq \beta \|y_{t+1} - x_t\| \cdot \|y_{t+1} - x_{t+1}\| \\
&\leq \frac{\beta}{2}\|y_{t+1} - x_t\|^2 + \frac{\beta}{2}\|y_{t+1} - x_{t+1}\|^2.
\end{aligned}
$$

Thus summing up these three terms and using that $\eta = \frac{\rho}{\beta}$ one gets

$$
f(y_{t+1}) - f(x) \leq \frac{D_\Phi(x, x_t) - D_\Phi(x, x_{t+1})}{\eta}.
$$

The proof is concluded with straightforward computations. $\qquad\square$

4.6 The vector field point of view on MD, DA, and MP

In this section we consider a mirror map Φ that satisfies the assumptions from Theorem 4.2.

By inspecting the proof of Theorem 4.2 one can see that for arbitrary vectors $g_1, \ldots, g_t \in \mathbb{R}^n$ the mirror descent strategy described by (4.2) or (4.3) (or alternatively by (4.5)) satisfies for any $x \in \mathcal{X} \cap \mathcal{D}$,

$$
\sum_{s=1}^{t} g_s^\top (x_s - x) \leq \frac{R^2}{\eta} + \frac{\eta}{2\rho} \sum_{s=1}^{t} \|g_s\|_*^2. \qquad (4.10)
$$

The observation that the sequence of vectors (g_s) does not have to come from the subgradients of a *fixed* function f is the starting point for the theory of online learning, see Bubeck [2011] for more details. In this

monograph we will use this observation to generalize mirror descent to saddle point calculations as well as stochastic settings. We note that we could also use dual averaging (defined by (4.6)) which satisfies

$$\sum_{s=1}^{t} g_s^\top (x_s - x) \leq \frac{R^2}{\eta} + \frac{2\eta}{\rho} \sum_{s=1}^{t} \|g_s\|_*^2.$$

In order to generalize mirror prox we simply replace the gradient ∇f by an arbitrary vector field $g : \mathcal{X} \to \mathbb{R}^n$ which yields the following equations:

$$\nabla \Phi(y'_{t+1}) = \nabla \Phi(x_t) - \eta g(x_t),$$
$$y_{t+1} \in \operatorname*{argmin}_{x \in \mathcal{X} \cap \mathcal{D}} D_\Phi(x, y'_{t+1}),$$
$$\nabla \Phi(x'_{t+1}) = \nabla \Phi(x_t) - \eta g(y_{t+1}),$$
$$x_{t+1} \in \operatorname*{argmin}_{x \in \mathcal{X} \cap \mathcal{D}} D_\Phi(x, x'_{t+1}).$$

Under the assumption that the vector field is β-Lipschitz w.r.t. $\|\cdot\|$, i.e., $\|g(x) - g(y)\|_* \leq \beta \|x - y\|$ one obtains with $\eta = \frac{\rho}{\beta}$

$$\sum_{s=1}^{t} g(y_{s+1})^\top (y_{s+1} - x) \leq \frac{\beta R^2}{\rho}. \tag{4.11}$$

5

Beyond the black-box model

In the black-box model non-smoothness dramatically deteriorates the rate of convergence of first order methods from $1/t^2$ to $1/\sqrt{t}$. However, as we already pointed out in Section 1.5, we (almost) always know the function to be optimized *globally*. In particular the "source" of non-smoothness can often be identified. For instance the LASSO objective (see Section 1.1) is non-smooth, but it is a sum of a smooth part (the least squares fit) and a *simple* non-smooth part (the ℓ_1-norm). Using this specific structure we will propose in Section 5.1 a first order method with a $1/t^2$ convergence rate, despite the non-smoothness. In Section 5.2 we consider another type of non-smoothness that can effectively be overcome, where the function is the maximum of smooth functions. Finally we conclude this chapter with a concise description of interior point methods, for which the structural assumption is made on the constraint set rather than on the objective function.

5.1   Sum of a smooth and a simple non-smooth term

We consider here the following problem[1]:

$$\min_{x \in \mathbb{R}^n} f(x) + g(x),$$

where f is convex and β-smooth, and g is convex. We assume that f can be accessed through a first order oracle, and that g is known and "simple". What we mean by simplicity will be clear from the description of the algorithm. For instance a separable function, that is $g(x) = \sum_{i=1}^n g_i(x(i))$, will be considered as simple. The prime example being $g(x) = \|x\|_1$. This section is inspired from Beck and Teboulle [2009] (see also Nesterov [2007], Wright et al. [2009]).

ISTA (Iterative Shrinkage-Thresholding Algorithm)

Recall that gradient descent on the smooth function f can be written as (see (4.5))

$$x_{t+1} = \operatorname*{argmin}_{x \in \mathbb{R}^n} \eta \nabla f(x_t)^\top x + \frac{1}{2}\|x - x_t\|_2^2.$$

Here one wants to minimize $f + g$, and g is assumed to be known and "simple". Thus it seems quite natural to consider the following update rule, where only f is locally approximated with a first order oracle:

$$
\begin{aligned}
x_{t+1} &= \operatorname*{argmin}_{x \in \mathbb{R}^n} \eta(g(x) + \nabla f(x_t)^\top x) + \frac{1}{2}\|x - x_t\|_2^2 \\
&= \operatorname*{argmin}_{x \in \mathbb{R}^n} \ g(x) + \frac{1}{2\eta}\|x - (x_t - \eta \nabla f(x_t))\|_2^2. \quad (5.1)
\end{aligned}
$$

The algorithm described by the above iteration is known as ISTA (Iterative Shrinkage-Thresholding Algorithm). In terms of convergence rate it is easy to show that ISTA has the same convergence rate on $f + g$ as gradient descent on f. More precisely with $\eta = \frac{1}{\beta}$ one has

$$f(x_t) + g(x_t) - (f(x^*) + g(x^*)) \leq \frac{\beta\|x_1 - x^*\|_2^2}{2t}.$$

---

[1]We restrict to unconstrained minimization for sake of simplicity. One can extend the discussion to constrained minimization by using ideas from Section 3.2.

This improved convergence rate over a subgradient descent directly on $f + g$ comes at a price: in general (5.1) may be a difficult optimization problem by itself, and this is why one needs to assume that g is simple. For instance if g can be written as $g(x) = \sum_{i=1}^{n} g_i(x(i))$ then one can compute x_{t+1} by solving n convex problems in dimension 1. In the case where $g(x) = \lambda \|x\|_1$ this one-dimensional problem is given by:

$$\min_{x \in \mathbb{R}} \; \lambda |x| + \frac{1}{2\eta}(x - x_0)^2, \text{ where } x_0 \in \mathbb{R}.$$

Elementary computations shows that this problem has an analytical solution given by $\tau_{\lambda\eta}(x_0)$, where τ is the shrinkage operator (hence the name ISTA), defined by

$$\tau_\alpha(x) = (|x| - \alpha)_+ \text{sign}(x).$$

Much more is known about (5.1) (which is called the *proximal operator* of g), and in fact entire monographs have been written about this equation, see e.g. Parikh and Boyd [2013], Bach et al. [2012].

FISTA (Fast ISTA)

An obvious idea is to combine Nesterov's accelerated gradient descent (which results in a $1/t^2$ rate to optimize f) with ISTA. This results in FISTA (Fast ISTA) which is described as follows. Let

$$\lambda_0 = 0, \; \lambda_t = \frac{1 + \sqrt{1 + 4\lambda_{t-1}^2}}{2}, \text{ and } \gamma_t = \frac{1 - \lambda_t}{\lambda_{t+1}}.$$

Let $x_1 = y_1$ an arbitrary initial point, and

$$
\begin{aligned}
y_{t+1} &= \text{argmin}_{x \in \mathbb{R}^n} \; g(x) + \frac{\beta}{2}\|x - (x_t - \frac{1}{\beta}\nabla f(x_t))\|_2^2, \\
x_{t+1} &= (1 - \gamma_t)y_{t+1} + \gamma_t y_t.
\end{aligned}
$$

Again it is easy show that the rate of convergence of FISTA on $f + g$ is similar to the one of Nesterov's accelerated gradient descent on f, more precisely:

$$f(y_t) + g(y_t) - (f(x^*) + g(x^*)) \le \frac{2\beta\|x_1 - x^*\|^2}{t^2}.$$

CMD and RDA

ISTA and FISTA assume smoothness in the Euclidean metric. Quite naturally one can also use these ideas in a non-Euclidean setting. Starting from (4.5) one obtains the CMD (Composite Mirror Descent) algorithm of Duchi et al. [2010], while with (4.6) one obtains the RDA (Regularized Dual Averaging) of Xiao [2010]. We refer to these papers for more details.

5.2 Smooth saddle-point representation of a non-smooth function

Quite often the non-smoothness of a function f comes from a max operation. More precisely non-smooth functions can often be represented as

$$f(x) = \max_{1 \leq i \leq m} f_i(x), \tag{5.2}$$

where the functions f_i are smooth. This was the case for instance with the function we used to prove the black-box lower bound $1/\sqrt{t}$ for non-smooth optimization in Theorem 3.13. We will see now that by using this structural representation one can in fact attain a rate of $1/t$. This was first observed in Nesterov [2004b] who proposed the Nesterov's smoothing technique. Here we will present the alternative method of Nemirovski [2004a] which we find more transparent (yet another version is the Chambolle-Pock algorithm, see Chambolle and Pock [2011]). Most of what is described in this section can be found in Juditsky and Nemirovski [2011a,b].

In the next subsection we introduce the more general problem of saddle point computation. We then proceed to apply a modified version of mirror descent to this problem, which will be useful both in Chapter 6 and also as a warm-up for the more powerful modified mirror prox that we introduce next.

5.2.1 Saddle point computation

Let $\mathcal{X} \subset \mathbb{R}^n$, $\mathcal{Y} \subset \mathbb{R}^m$ be compact and convex sets. Let $\varphi : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ be a continuous function, such that $\varphi(\cdot, y)$ is convex and $\varphi(x, \cdot)$ is

concave. We write $g_{\mathcal{X}}(x, y)$ (respectively $g_{\mathcal{Y}}(x, y)$) for an element of $\partial_x \varphi(x, y)$ (respectively $\partial_y(-\varphi(x, y))$). We are interested in computing

$$\min_{x \in \mathcal{X}} \max_{y \in \mathcal{Y}} \varphi(x, y).$$

By Sion's minimax theorem there exists a pair $(x^*, y^*) \in \mathcal{X} \times \mathcal{Y}$ such that

$$\varphi(x^*, y^*) = \min_{x \in \mathcal{X}} \max_{y \in \mathcal{Y}} \varphi(x, y) = \max_{y \in \mathcal{Y}} \min_{x \in \mathcal{X}} \varphi(x, y).$$

We will explore algorithms that produce a candidate pair of solutions $(\widetilde{x}, \widetilde{y}) \in \mathcal{X} \times \mathcal{Y}$. The quality of $(\widetilde{x}, \widetilde{y})$ is evaluated through the so-called duality gap[2]

$$\max_{y \in \mathcal{Y}} \varphi(\widetilde{x}, y) - \min_{x \in \mathcal{X}} \varphi(x, \widetilde{y}).$$

The key observation is that the duality gap can be controlled similarly to the suboptimality gap $f(x) - f(x^*)$ in a simple convex optimization problem. Indeed for any $(x, y) \in \mathcal{X} \times \mathcal{Y}$,

$$\varphi(\widetilde{x}, \widetilde{y}) - \varphi(x, \widetilde{y}) \leq g_{\mathcal{X}}(\widetilde{x}, \widetilde{y})^\top (\widetilde{x} - x),$$

and

$$-\varphi(\widetilde{x}, \widetilde{y}) - (-\varphi(\widetilde{x}, y)) \leq g_{\mathcal{Y}}(\widetilde{x}, \widetilde{y})^\top (\widetilde{y} - y).$$

In particular, using the notation $z = (x, y) \in \mathcal{Z} := \mathcal{X} \times \mathcal{Y}$ and $g(z) = (g_{\mathcal{X}}(x, y), g_{\mathcal{Y}}(x, y))$ we just proved

$$\max_{y \in \mathcal{Y}} \varphi(\widetilde{x}, y) - \min_{x \in \mathcal{X}} \varphi(x, \widetilde{y}) \leq g(\widetilde{z})^\top (\widetilde{z} - z), \qquad (5.3)$$

for some $z \in \mathcal{Z}$. In view of the vector field point of view developed in Section 4.6 this suggests to do a mirror descent in the $\mathcal{Z}$-space with the vector field $g : \mathcal{Z} \to \mathbb{R}^n \times \mathbb{R}^m$.

We will assume in the next subsections that $\mathcal{X}$ is equipped with a mirror map $\Phi_{\mathcal{X}}$ (defined on $\mathcal{D}_{\mathcal{X}}$) which is 1-strongly convex w.r.t. a norm $\| \cdot \|_{\mathcal{X}}$ on $\mathcal{X} \cap \mathcal{D}_{\mathcal{X}}$. We denote $R_{\mathcal{X}}^2 = \sup_{x \in \mathcal{X}} \Phi(x) - \min_{x \in \mathcal{X}} \Phi(x)$. We define similar quantities for the space $\mathcal{Y}$.

---

[2]Observe that the duality gap is the sum of the primal gap $\max_{y \in \mathcal{Y}} \varphi(\widetilde{x}, y) - \varphi(x^*, y^*)$ and the dual gap $\varphi(x^*, y^*) - \min_{x \in \mathcal{X}} \varphi(x, \widetilde{y})$.

5.2.2 Saddle Point Mirror Descent (SP-MD)

We consider here mirror descent on the space $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$ with the mirror map $\Phi(z) = a\Phi_{\mathcal{X}}(x) + b\Phi_{\mathcal{Y}}(y)$ (defined on $\mathcal{D} = \mathcal{D}_{\mathcal{X}} \times \mathcal{D}_{\mathcal{Y}}$), where $a, b \in \mathbb{R}_+$ are to be defined later, and with the vector field $g : \mathcal{Z} \to \mathbb{R}^n \times \mathbb{R}^m$ defined in the previous subsection. We call the resulting algorithm SP-MD (Saddle Point Mirror Descent). It can be described succinctly as follows.

Let $z_1 \in \operatorname{argmin}_{z \in \mathcal{Z} \cap \mathcal{D}} \Phi(z)$. Then for $t \geq 1$, let

$$z_{t+1} \in \operatorname*{argmin}_{z \in \mathcal{Z} \cap \mathcal{D}} \; \eta g_t^\top z + D_\Phi(z, z_t),$$

where $g_t = (g_{\mathcal{X},t}, g_{\mathcal{Y},t})$ with $g_{\mathcal{X},t} \in \partial_x \varphi(x_t, y_t)$ and $g_{\mathcal{Y},t} \in \partial_y(-\varphi(x_t, y_t))$.

Theorem 5.1. Assume that $\varphi(\cdot, y)$ is $L_{\mathcal{X}}$-Lipschitz w.r.t. $\|\cdot\|_{\mathcal{X}}$, that is $\|g_{\mathcal{X}}(x, y)\|_{\mathcal{X}}^* \leq L_{\mathcal{X}}, \forall (x, y) \in \mathcal{X} \times \mathcal{Y}$. Similarly assume that $\varphi(x, \cdot)$ is $L_{\mathcal{Y}}$-Lipschitz w.r.t. $\|\cdot\|_{\mathcal{Y}}$. Then SP-MD with $a = \frac{L_{\mathcal{X}}}{R_{\mathcal{X}}}$, $b = \frac{L_{\mathcal{Y}}}{R_{\mathcal{Y}}}$, and $\eta = \sqrt{\frac{2}{t}}$ satisfies

$$\max_{y \in \mathcal{Y}} \varphi\left(\frac{1}{t}\sum_{s=1}^t x_s, y\right) - \min_{x \in \mathcal{X}} \varphi\left(x, \frac{1}{t}\sum_{s=1}^t y_s\right) \leq (R_{\mathcal{X}} L_{\mathcal{X}} + R_{\mathcal{Y}} L_{\mathcal{Y}})\sqrt{\frac{2}{t}}.$$

Proof. First we endow $\mathcal{Z}$ with the norm $\|\cdot\|_{\mathcal{Z}}$ defined by

$$\|z\|_{\mathcal{Z}} = \sqrt{a\|x\|_{\mathcal{X}}^2 + b\|y\|_{\mathcal{Y}}^2}.$$

It is immediate that Φ is 1-strongly convex with respect to $\|\cdot\|_{\mathcal{Z}}$ on $\mathcal{Z} \cap \mathcal{D}$. Furthermore one can easily check that

$$\|z\|_{\mathcal{Z}}^* = \sqrt{\frac{1}{a}\left(\|x\|_{\mathcal{X}}^*\right)^2 + \frac{1}{b}\left(\|y\|_{\mathcal{Y}}^*\right)^2},$$

and thus the vector field (g_t) used in the SP-MD satisfies:

$$\|g_t\|_{\mathcal{Z}}^* \leq \sqrt{\frac{L_{\mathcal{X}}^2}{a} + \frac{L_{\mathcal{Y}}^2}{b}}.$$

Using (4.10) together with (5.3) and the values of a, b and η concludes the proof. $\qquad\square$

5.2.3   Saddle Point Mirror Prox (SP-MP)

We now consider the most interesting situation in the context of this chapter, where the function φ is smooth. Precisely we say that φ is $(\beta_{11}, \beta_{12}, \beta_{22}, \beta_{21})$-smooth if for any $x, x' \in \mathcal{X}, y, y' \in \mathcal{Y}$,

$$
\begin{aligned}
\|\nabla_x \varphi(x, y) - \nabla_x \varphi(x', y)\|_{\mathcal{X}}^* &\le \beta_{11} \|x - x'\|_{\mathcal{X}}, \\
\|\nabla_x \varphi(x, y) - \nabla_x \varphi(x, y')\|_{\mathcal{X}}^* &\le \beta_{12} \|y - y'\|_{\mathcal{Y}}, \\
\|\nabla_y \varphi(x, y) - \nabla_y \varphi(x, y')\|_{\mathcal{Y}}^* &\le \beta_{22} \|y - y'\|_{\mathcal{Y}}, \\
\|\nabla_y \varphi(x, y) - \nabla_y \varphi(x', y)\|_{\mathcal{Y}}^* &\le \beta_{21} \|x - x'\|_{\mathcal{X}},
\end{aligned}
$$

This will imply the Lipschitzness of the vector field $g : \mathcal{Z} \to \mathbb{R}^n \times \mathbb{R}^m$ under the appropriate norm. Thus we use here mirror prox on the space $\mathcal{Z}$ with the mirror map $\Phi(z) = a\Phi_{\mathcal{X}}(x) + b\Phi_{\mathcal{Y}}(y)$ and the vector field g. The resulting algorithm is called SP-MP (Saddle Point Mirror Prox) and we can describe it succintly as follows.

Let $z_1 \in \operatorname{argmin}_{z \in \mathcal{Z} \cap \mathcal{D}} \Phi(z)$. Then for $t \ge 1$, let $z_t = (x_t, y_t)$ and $w_t = (u_t, v_t)$ be defined by

$$
\begin{aligned}
w_{t+1} &= \operatorname*{argmin}_{z \in \mathcal{Z} \cap \mathcal{D}} \eta(\nabla_x \varphi(x_t, y_t), -\nabla_y \varphi(x_t, y_t))^\top z + D_\Phi(z, z_t) \\
z_{t+1} &= \operatorname*{argmin}_{z \in \mathcal{Z} \cap \mathcal{D}} \eta(\nabla_x \varphi(u_{t+1}, v_{t+1}), -\nabla_y \varphi(u_{t+1}, v_{t+1}))^\top z + D_\Phi(z, z_t).
\end{aligned}
$$

Theorem 5.2. Assume that φ is $(\beta_{11}, \beta_{12}, \beta_{22}, \beta_{21})$-smooth. Then SP-MP with $a = \frac{1}{R_{\mathcal{X}}^2}$, $b = \frac{1}{R_{\mathcal{Y}}^2}$, and $\eta = 1/\left(2\max\left(\beta_{11} R_{\mathcal{X}}^2, \beta_{22} R_{\mathcal{Y}}^2, \beta_{12} R_{\mathcal{X}} R_{\mathcal{Y}}, \beta_{21} R_{\mathcal{X}} R_{\mathcal{Y}}\right)\right)$ satisfies

$$
\max_{y \in \mathcal{Y}} \varphi\left(\frac{1}{t}\sum_{s=1}^t u_{s+1}, y\right) - \min_{x \in \mathcal{X}} \varphi\left(x, \frac{1}{t}\sum_{s=1}^t v_{s+1}\right)
$$
$$
\le \max\left(\beta_{11} R_{\mathcal{X}}^2, \beta_{22} R_{\mathcal{Y}}^2, \beta_{12} R_{\mathcal{X}} R_{\mathcal{Y}}, \beta_{21} R_{\mathcal{X}} R_{\mathcal{Y}}\right) \frac{4}{t}.
$$

Proof. In light of the proof of Theorem 5.1 and (4.11) it clearly suffices to show that the vector field $g(z) = (\nabla_x \varphi(x, y), -\nabla_y \varphi(x, y))$ is β-Lipschitz w.r.t. $\|z\|_{\mathcal{Z}} = \sqrt{\frac{1}{R_{\mathcal{X}}^2}\|x\|_{\mathcal{X}}^2 + \frac{1}{R_{\mathcal{Y}}^2}\|y\|_{\mathcal{Y}}^2}$ with $\beta = 2\max\left(\beta_{11} R_{\mathcal{X}}^2, \beta_{22} R_{\mathcal{Y}}^2, \beta_{12} R_{\mathcal{X}} R_{\mathcal{Y}}, \beta_{21} R_{\mathcal{X}} R_{\mathcal{Y}}\right)$. In other words one needs to show that

$$
\|g(z) - g(z')\|_{\mathcal{Z}}^* \le \beta \|z - z'\|_{\mathcal{Z}},
$$

which can be done with straightforward calculations (by introducing $g(x', y)$ and using the definition of smoothness for φ). $\square$

5.2.4 Applications

We investigate briefly three applications for SP-MD and SP-MP.

Minimizing a maximum of smooth functions

The problem (5.2) (when f has to minimized over $\mathcal{X}$) can be rewritten as

$$\min_{x \in \mathcal{X}} \max_{y \in \Delta_m} \vec{f}(x)^\top y,$$

where $\vec{f}(x) = (f_1(x), \ldots, f_m(x)) \in \mathbb{R}^m$. We assume that the functions f_i are L-Lipschtiz and β-smooth w.r.t. some norm $\|\cdot\|_{\mathcal{X}}$. Let us study the smoothness of $\varphi(x, y) = \vec{f}(x)^\top y$ when $\mathcal{X}$ is equipped with $\|\cdot\|_{\mathcal{X}}$ and Δ_m is equipped with $\|\cdot\|_1$. On the one hand $\nabla_y \varphi(x, y) = \vec{f}(x)$, in particular one immediately has $\beta_{22} = 0$, and furthermore

$$\|\vec{f}(x) - \vec{f}(x')\|_\infty \leq L\|x - x'\|_{\mathcal{X}},$$

that is $\beta_{21} = L$. On the other hand $\nabla_x \varphi(x, y) = \sum_{i=1}^m y_i \nabla f_i(x)$, and thus

$$\|\sum_{i=1}^m y(i)(\nabla f_i(x) - \nabla f_i(x'))\|_{\mathcal{X}}^* \leq \beta\|x - x'\|_{\mathcal{X}},$$

$$\|\sum_{i=1}^m (y(i) - y'(i))\nabla f_i(x)\|_{\mathcal{X}}^* \leq L\|y - y'\|_1,$$

that is $\beta_{11} = \beta$ and $\beta_{12} = L$. Thus using SP-MP with some mirror map on $\mathcal{X}$ and the negentropy on Δ_m (see the "simplex setup" in Section 4.3), one obtains an ε-optimal point of $f(x) = \max_{1 \leq i \leq m} f_i(x)$ in $O\left(\frac{\beta R_{\mathcal{X}}^2 + L R_{\mathcal{X}} \sqrt{\log(m)}}{\varepsilon}\right)$ iterations. Furthermore an iteration of SP-MP has a computational complexity of order of a step of mirror descent in $\mathcal{X}$ on the function $x \mapsto \sum_{i=1}^m y(i) f_i(x)$ (plus $O(m)$ for the update in the $\mathcal{Y}$-space).

Thus by using the structure of f we were able to obtain a much better rate than black-box procedures (which would have required $\Omega(1/\varepsilon^2)$ iterations as f is potentially non-smooth).

Matrix games

Let $A \in \mathbb{R}^{n \times m}$, we denote $\|A\|_{\max}$ for the maximal entry (in absolute value) of A, and $A_i \in \mathbb{R}^n$ for the ith column of A. We consider the problem of computing a Nash equilibrium for the zero-sum game corresponding to the loss matrix A, that is we want to solve

$$\min_{x \in \Delta_n} \max_{y \in \Delta_m} x^\top A y.$$

Here we equip both Δ_n and Δ_m with $\| \cdot \|_1$. Let $\varphi(x, y) = x^\top A y$. Using that $\nabla_x \varphi(x, y) = Ay$ and $\nabla_y \varphi(x, y) = A^\top x$ one immediately obtains $\beta_{11} = \beta_{22} = 0$. Furthermore since

$$\|A(y - y')\|_\infty = \|\sum_{i=1}^m (y(i) - y'(i)) A_i\|_\infty \le \|A\|_{\max} \|y - y'\|_1,$$

one also has $\beta_{12} = \beta_{21} = \|A\|_{\max}$. Thus SP-MP with the negentropy on both Δ_n and Δ_m attains an ε-optimal pair of mixed strategies with $O\left(\|A\|_{\max}\sqrt{\log(n)\log(m)}/\varepsilon\right)$ iterations. Furthermore the computational complexity of a step of SP-MP is dominated by the matrix-vector multiplications which are $O(nm)$. Thus overall the complexity of getting an ε-optimal Nash equilibrium with SP-MP is $O\left(\|A\|_{\max} nm \sqrt{\log(n)\log(m)}/\varepsilon\right)$.

Linear classification

Let $(\ell_i, A_i) \in \{-1, 1\} \times \mathbb{R}^n$, $i \in [m]$, be a data set that one wishes to separate with a linear classifier. That is one is looking for $x \in \mathrm{B}_{2,n}$ such that for all $i \in [m]$, $\mathrm{sign}(x^\top A_i) = \mathrm{sign}(\ell_i)$, or equivalently $\ell_i x^\top A_i > 0$. Clearly without loss of generality one can assume $\ell_i = 1$ for all $i \in [m]$ (simply replace A_i by $\ell_i A_i$). Let $A \in \mathbb{R}^{n \times m}$ be the matrix where the ith column is A_i. The problem of finding x with maximal margin can be written as

$$\max_{x \in \mathrm{B}_{2,n}} \min_{1 \le i \le m} A_i^\top x = \max_{x \in \mathrm{B}_{2,n}} \min_{y \in \Delta_m} x^\top A y. \tag{5.4}$$

Assuming that $\|A_i\|_2 \le B$, and using the calculations we did in Section 5.2.4, it is clear that $\varphi(x, y) = x^\top A y$ is $(0, B, 0, B)$-smooth with respect

to $\| \cdot \|_2$ on $B_{2,n}$ and $\| \cdot \|_1$ on Δ_m. This implies in particular that SP-MP with the Euclidean norm squared on $B_{2,n}$ and the negentropy on Δ_m will solve (5.4) in $O(B\sqrt{\log(m)}/\varepsilon)$ iterations. Again the cost of an iteration is dominated by the matrix-vector multiplications, which results in an overall complexity of $O(Bnm\sqrt{\log(m)}/\varepsilon)$ to find an ε-optimal solution to (5.4).

5.3 Interior point methods

We describe here interior point methods (IPM), a class of algorithms fundamentally different from what we have seen so far. The first algorithm of this type was described in Karmarkar [1984], but the theory we shall present was developed in Nesterov and Nemirovski [1994]. We follow closely the presentation given in [Chapter 4, Nesterov [2004a]]. Other useful references (in particular for the primal-dual IPM, which are the ones used in practice) include Renegar [2001], Nemirovski [2004b], Nocedal and Wright [2006].

IPM are designed to solve convex optimization problems of the form

$$\text{min. } c^\top x$$
$$\text{s.t. } x \in \mathcal{X},$$

with $c \in \mathbb{R}^n$, and $\mathcal{X} \subset \mathbb{R}^n$ convex and compact. Note that, at this point, the linearity of the objective is without loss of generality as minimizing a convex function f over $\mathcal{X}$ is equivalent to minimizing a linear objective over the epigraph of f (which is also a convex set). The structural assumption on $\mathcal{X}$ that one makes in IPM is that there exists a *self-concordant barrier* for $\mathcal{X}$ with an easily computable gradient and Hessian. The meaning of the previous sentence will be made precise in the next subsections. The importance of IPM stems from the fact that LPs and SDPs (see Section 1.5) satisfy this structural assumption.

5.3.1 The barrier method

We say that $F : \text{int}(\mathcal{X}) \to \mathbb{R}$ is a *barrier* for $\mathcal{X}$ if

$$F(x) \xrightarrow[x \to \partial \mathcal{X}]{} +\infty.$$

We will only consider strictly convex barriers. We extend the domain of definition of F to $\mathbb{R}^n$ with $F(x) = +\infty$ for $x \notin \text{int}(\mathcal{X})$. For $t \in \mathbb{R}_+$ let

$$x^*(t) \in \operatorname*{argmin}_{x \in \mathbb{R}^n} tc^\top x + F(x).$$

In the following we denote $F_t(x) := tc^\top x + F(x)$. In IPM the path $(x^*(t))_{t \in \mathbb{R}_+}$ is referred to as the *central path*. It seems clear that the central path eventually leads to the minimum x^* of the objective function $c^\top x$ on $\mathcal{X}$, precisely we will have

$$x^*(t) \xrightarrow[t \to +\infty]{} x^*.$$

The idea of the *barrier method* is to move along the central path by "boosting" a fast locally convergent algorithm, which we denote for the moment by $\mathcal{A}$, using the following scheme: Assume that one has computed $x^*(t)$, then one uses $\mathcal{A}$ initialized at $x^*(t)$ to compute $x^*(t')$ for some $t' > t$. There is a clear tension for the choice of t', on the one hand t' should be large in order to make as much progress as possible on the central path, but on the other hand $x^*(t)$ needs to be close enough to $x^*(t')$ so that it is in the basin of fast convergence for $\mathcal{A}$ when run on $F_{t'}$.

IPM follows the above methodology with $\mathcal{A}$ being *Newton's method*. Indeed as we will see in the next subsection, Newton's method has a quadratic convergence rate, in the sense that if initialized close enough to the optimum it attains an ε-optimal point in $\log \log(1/\varepsilon)$ iterations! Thus we now have a clear plan to make these ideas formal and analyze the iteration complexity of IPM:

1. First we need to describe precisely the region of fast convergence for Newton's method. This will lead us to define self-concordant functions, which are "natural" functions for Newton's method.

2. Then we need to evaluate precisely how much larger t' can be compared to t, so that $x^*(t)$ is still in the region of fast convergence of Newton's method when optimizing the function $F_{t'}$ with $t' > t$. This will lead us to define ν-self concordant barriers.

3. How do we get close to the central path in the first place? Is it possible to compute $x^*(0) = \mathrm{argmin}_{x \in \mathbb{R}^n} F(x)$ (the so-called analytical center of $\mathcal{X}$)?

5.3.2 Traditional analysis of Newton's method

We start by describing Newton's method together with its standard analysis showing the quadratic convergence rate when initialized close enough to the optimum. In this subsection we denote $\| \cdot \|$ for both the Euclidean norm on $\mathbb{R}^n$ and the operator norm on matrices (in particular $\|Ax\| \leq \|A\| \cdot \|x\|$).

Let $f : \mathbb{R}^n \to \mathbb{R}$ be a C^2 function. Using a Taylor's expansion of f around x one obtains

$$f(x + h) = f(x) + h^\top \nabla f(x) + \frac{1}{2} h^\top \nabla^2 f(x) h + o(\|h\|^2).$$

Thus, starting at x, in order to minimize f it seems natural to move in the direction h that minimizes

$$h^\top \nabla f(x) + \frac{1}{2} h^\top \nabla f^2(x) h.$$

If $\nabla^2 f(x)$ is positive definite then the solution to this problem is given by $h = -[\nabla^2 f(x)]^{-1} \nabla f(x)$. Newton's method simply iterates this idea: starting at some point $x_0 \in \mathbb{R}^n$, it iterates for $k \geq 0$ the following equation:

$$x_{k+1} = x_k - [\nabla^2 f(x_k)]^{-1} \nabla f(x_k).$$

While this method can have an arbitrarily bad behavior in general, if started close enough to a strict local minimum of f, it can have a very fast convergence:

Theorem 5.3. Assume that f has a Lipschitz Hessian, that is $\|\nabla^2 f(x) - \nabla^2 f(y)\| \leq M \|x - y\|$. Let x^* be local minimum of f with strictly positive Hessian, that is $\nabla^2 f(x^*) \succeq \mu I_n$, $\mu > 0$. Suppose that the initial starting point x_0 of Newton's method is such that

$$\|x_0 - x^*\| \leq \frac{\mu}{2M}.$$

Then Newton's method is well-defined and converges to x^* at a quadratic rate:

$$\|x_{k+1} - x^*\| \leq \frac{M}{\mu} \|x_k - x^*\|^2.$$

Proof. We use the following simple formula, for $x, h \in \mathbb{R}^n$,

$$\int_0^1 \nabla^2 f(x + sh) \, h \, ds = \nabla f(x + h) - \nabla f(x).$$

Now note that $\nabla f(x^*) = 0$, and thus with the above formula one obtains

$$\nabla f(x_k) = \int_0^1 \nabla^2 f(x^* + s(x_k - x^*)) \, (x_k - x^*) \, ds,$$

which allows us to write:

$$
\begin{aligned}
& x_{k+1} - x^* \\
&= x_k - x^* - [\nabla^2 f(x_k)]^{-1} \nabla f(x_k) \\
&= x_k - x^* - [\nabla^2 f(x_k)]^{-1} \int_0^1 \nabla^2 f(x^* + s(x_k - x^*)) \, (x_k - x^*) \, ds \\
&= [\nabla^2 f(x_k)]^{-1} \int_0^1 [\nabla^2 f(x_k) - \nabla^2 f(x^* + s(x_k - x^*))] \, (x_k - x^*) \, ds.
\end{aligned}
$$

In particular one has

$$
\begin{aligned}
& \|x_{k+1} - x^*\| \\
&\leq \|[\nabla^2 f(x_k)]^{-1}\| \\
&\quad \times \left(\int_0^1 \|\nabla^2 f(x_k) - \nabla^2 f(x^* + s(x_k - x^*))\| \, ds \right) \|x_k - x^*\|.
\end{aligned}
$$

Using the Lipschitz property of the Hessian one immediately obtains that

$$\left(\int_0^1 \|\nabla^2 f(x_k) - \nabla^2 f(x^* + s(x_k - x^*))\| \, ds \right) \leq \frac{M}{2} \|x_k - x^*\|.$$

Using again the Lipschitz property of the Hessian (note that $\|A - B\| \leq s \Leftrightarrow s\mathrm{I}_n \succeq A - B \succeq -s\mathrm{I}_n$), the hypothesis on x^*, and an induction hypothesis that $\|x_k - x^*\| \leq \frac{\mu}{2M}$, one has

$$\nabla^2 f(x_k) \succeq \nabla^2 f(x^*) - M\|x_k - x^*\|\mathrm{I}_n \succeq (\mu - M\|x_k - x^*\|)\mathrm{I}_n \succeq \frac{\mu}{2}\mathrm{I}_n,$$

which concludes the proof. $\qquad\square$

5.3.3   Self-concordant functions

Before giving the definition of self-concordant functions let us try to get some insight into the "geometry" of Newton's method. Let A be a $n \times n$ non-singular matrix. We look at a Newton step on the functions $f : x \mapsto f(x)$ and $\varphi : y \mapsto f(A^{-1}y)$, starting respectively from x and $y = Ax$, that is:

$$x^+ = x - [\nabla^2 f(x)]^{-1} \nabla f(x), \text{ and } y^+ = y - [\nabla^2 \varphi(y)]^{-1} \nabla \varphi(y).$$

By using the following simple formulas

$$\nabla(x \mapsto f(Ax)) = A^\top \nabla f(Ax), \text{ and } \nabla^2(x \mapsto f(Ax)) = A^\top \nabla^2 f(Ax)A.$$

it is easy to show that

$$y^+ = Ax^+.$$

In other words Newton's method will follow the same trajectory in the "x-space" and in the "y-space" (the image through A of the x-space), that is Newton's method is *affine invariant*. Observe that this property is not shared by the methods described in Chapter 3 (except for the conditional gradient descent).

   The affine invariance of Newton's method casts some concerns on the assumptions of the analysis in Section 5.3.2. Indeed the assumptions are all in terms of the canonical inner product in $\mathbb{R}^n$. However we just showed that the method itself does not depend on the choice of the inner product (again this is not true for first order methods). Thus one would like to derive a result similar to Theorem 5.3 without any reference to a prespecified inner product. The idea of self-concordance is to modify the Lipschitz assumption on the Hessian to achieve this goal.

   Assume from now on that f is C^3, and let $\nabla^3 f(x) : \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$ be the third order differential operator. The Lipschitz assumption on the Hessian in Theorem 5.3 can be written as:

$$\nabla^3 f(x)[h, h, h] \leq M \|h\|_2^3.$$

The issue is that this inequality depends on the choice of an inner product. More importantly it is easy to see that a convex function which

goes to infinity on a compact set simply cannot satisfy the above inequality. A natural idea to try fix these issues is to replace the Euclidean metric on the right hand side by the metric given by the function f itself at x, that is:

$$\|h\|_x = \sqrt{h^\top \nabla^2 f(x) h}.$$

Observe that to be clear one should rather use the notation $\|\cdot\|_{x,f}$, but since f will always be clear from the context we stick to $\|\cdot\|_x$.

Definition 5.1. Let $\mathcal{X}$ be a convex set with non-empty interior, and f a C^3 convex function defined on $\text{int}(\mathcal{X})$. Then f is self-concordant (with constant M) if for all $x \in \text{int}(\mathcal{X}), h \in \mathbb{R}^n$,

$$\nabla^3 f(x)[h, h, h] \leq M \|h\|_x^3.$$

We say that f is standard self-concordant if f is self-concordant with constant $M = 2$.

An easy consequence of the definition is that a self-concordant function is a barrier for the set $\mathcal{X}$, see [Theorem 4.1.4, Nesterov [2004a]]. The main example to keep in mind of a standard self-concordant function is $f(x) = -\log x$ for $x > 0$. The next definition will be key in order to describe the region of quadratic convergence for Newton's method on self-concordant functions.

Definition 5.2. Let f be a standard self-concordant function on $\mathcal{X}$. For $x \in \text{int}(\mathcal{X})$, we say that $\lambda_f(x) = \|\nabla f(x)\|_x^*$ is the *Newton decrement* of f at x.

An important inequality is that for x such that $\lambda_f(x) < 1$, and $x^* = \operatorname{argmin} f(x)$, one has

$$\|x - x^*\|_x \leq \frac{\lambda_f(x)}{1 - \lambda_f(x)}, \tag{5.5}$$

see [Equation 4.1.18, Nesterov [2004a]]. We state the next theorem without a proof, see also [Theorem 4.1.14, Nesterov [2004a]].

Theorem 5.4. Let f be a standard self-concordant function on $\mathcal{X}$, and $x \in \text{int}(\mathcal{X})$ such that $\lambda_f(x) \leq 1/4$, then

$$\lambda_f\left(x - [\nabla^2 f(x)]^{-1}\nabla f(x)\right) \leq 2\lambda_f(x)^2.$$

In other words the above theorem states that, if initialized at a point x_0 such that $\lambda_f(x_0) \leq 1/4$, then Newton's iterates satisfy $\lambda_f(x_{k+1}) \leq 2\lambda_f(x_k)^2$. Thus, Newton's region of quadratic convergence for self-concordant functions can be described as a "Newton decrement ball" $\{x : \lambda_f(x) \leq 1/4\}$. In particular by taking the barrier to be a self-concordant function we have now resolved Step (1) of the plan described in Section 5.3.1.

5.3.4 ν-self-concordant barriers

We deal here with Step (2) of the plan described in Section 5.3.1. Given Theorem 5.4 we want t' to be as large as possible and such that

$$\lambda_{F_{t'}}(x^*(t)) \leq 1/4. \tag{5.6}$$

Since the Hessian of $F_{t'}$ is the Hessian of F, one has

$$\lambda_{F_{t'}}(x^*(t)) = \|t'c + \nabla F(x^*(t))\|_{x^*(t)}^*.$$

Observe that, by first order optimality, one has $tc + \nabla F(x^*(t)) = 0$, which yields

$$\lambda_{F_{t'}}(x^*(t)) = (t' - t)\|c\|_{x^*(t)}^*. \tag{5.7}$$

Thus taking

$$t' = t + \frac{1}{4\|c\|_{x^*(t)}^*} \tag{5.8}$$

immediately yields (5.6). In particular with the value of t' given in (5.8) the Newton's method on $F_{t'}$ initialized at $x^*(t)$ will converge quadratically fast to $x^*(t')$.

It remains to verify that by iterating (5.8) one obtains a sequence diverging to infinity, and to estimate the rate of growth. Thus one needs to control $\|c\|_{x^*(t)}^* = \frac{1}{t}\|\nabla F(x^*(t))\|_{x^*(t)}^*$. Luckily there is a natural class of functions for which one can control $\|\nabla F(x)\|_x^*$ uniformly over x. This is the set of functions such that

$$\nabla^2 F(x) \succeq \frac{1}{\nu}\nabla F(x)[\nabla F(x)]^\top. \tag{5.9}$$

Indeed in that case one has:

$$
\begin{aligned}
\|\nabla F(x)\|_x^* &= \sup_{h:h^\top \nabla F^2(x)h \leq 1} \nabla F(x)^\top h \\
&\leq \sup_{h:h^\top \left(\frac{1}{\nu}\nabla F(x)[\nabla F(x)]^\top\right)h \leq 1} \nabla F(x)^\top h \\
&= \sqrt{\nu}.
\end{aligned}
$$

Thus a safe choice to increase the penalization parameter is $t' = \left(1 + \frac{1}{4\sqrt{\nu}}\right)t$. Note that the condition (5.9) can also be written as the fact that the function F is $\frac{1}{\nu}$-exp-concave, that is $x \mapsto \exp(-\frac{1}{\nu}F(x))$ is concave. We arrive at the following definition.

Definition 5.3. F is a ν-self-concordant barrier if it is a standard self-concordant function, and it is $\frac{1}{\nu}$-exp-concave.

Again the canonical example is the logarithmic function, $x \mapsto -\log x$, which is a 1-self-concordant barrier for the set $\mathbb{R}_+$. We state the next theorem without a proof (see Bubeck and Eldan [2014] for more on this result).

Theorem 5.5. Let $\mathcal{X} \subset \mathbb{R}^n$ be a closed convex set with non-empty interior. There exists F which is a $(c\, n)$-self-concordant barrier for $\mathcal{X}$ (where c is some universal constant).

A key property of ν-self-concordant barriers is the following inequality:

$$
c^\top x^*(t) - \min_{x \in \mathcal{X}} c^\top x \leq \frac{\nu}{t}, \tag{5.10}
$$

see [Equation (4.2.17), Nesterov [2004a]]. More generally using (5.10) together with (5.5) one obtains

$$
\begin{aligned}
c^\top y - \min_{x \in \mathcal{X}} c^\top x &\leq \frac{\nu}{t} + c^\top(y - x^*(t)) \\
&= \frac{\nu}{t} + \frac{1}{t}(\nabla F_t(y) - \nabla F(y))^\top(y - x^*(t)) \\
&\leq \frac{\nu}{t} + \frac{1}{t}\|\nabla F_t(y) - \nabla F(y)\|_y^* \cdot \|y - x^*(t)\|_y \\
&\leq \frac{\nu}{t} + \frac{1}{t}(\lambda_{F_t}(y) + \sqrt{\nu})\frac{\lambda_{F_t}(y)}{1 - \lambda_{F_t}(y)} \tag{5.11}
\end{aligned}
$$

In the next section we describe a precise algorithm based on the ideas we developed above. As we will see one cannot ensure to be exactly on the central path, and thus it is useful to generalize the identity (5.7) for a point x close to the central path. We do this as follows:

$$
\begin{aligned}
\lambda_{F_{t'}}(x) &= \|t'c + \nabla F(x)\|_x^* \\
&= \|(t'/t)(tc + \nabla F(x)) + (1 - t'/t)\nabla F(x)\|_x^* \\
&\leq \frac{t'}{t}\lambda_{F_t}(x) + \left(\frac{t'}{t} - 1\right)\sqrt{\nu}. \tag{5.12}
\end{aligned}
$$

5.3.5   Path-following scheme

We can now formally describe and analyze the most basic IPM called the *path-following scheme*. Let F be ν-self-concordant barrier for $\mathcal{X}$. Assume that one can find x_0 such that $\lambda_{F_{t_0}}(x_0) \leq 1/4$ for some small value $t_0 > 0$ (we describe a method to find x_0 at the end of this subsection). Then for $k \geq 0$, let

$$
\begin{aligned}
t_{k+1} &= \left(1 + \frac{1}{13\sqrt{\nu}}\right) t_k, \\
x_{k+1} &= x_k - [\nabla^2 F(x_k)]^{-1}(t_{k+1}c + \nabla F(x_k)).
\end{aligned}
$$

The next theorem shows that after $O\left(\sqrt{\nu}\log\frac{\nu}{t_0\varepsilon}\right)$ iterations of the path-following scheme one obtains an ε-optimal point.

Theorem 5.6. The path-following scheme described above satisfies

$$
c^\top x_k - \min_{x \in \mathcal{X}} c^\top x \leq \frac{2\nu}{t_0}\exp\left(-\frac{k}{1 + 13\sqrt{\nu}}\right).
$$

Proof. We show that the iterates $(x_k)_{k \geq 0}$ remain close to the central path $(x^*(t_k))_{k \geq 0}$. Precisely one can easily prove by induction that

$$
\lambda_{F_{t_k}}(x_k) \leq 1/4.
$$

Indeed using Theorem 5.4 and equation (5.12) one immediately obtains

$$
\begin{aligned}
\lambda_{F_{t_{k+1}}}(x_{k+1}) &\leq 2\lambda_{F_{t_{k+1}}}(x_k)^2 \\
&\leq 2\left(\frac{t_{k+1}}{t_k}\lambda_{F_{t_k}}(x_k) + \left(\frac{t_{k+1}}{t_k} - 1\right)\sqrt{\nu}\right)^2 \\
&\leq 1/4,
\end{aligned}
$$

where we used in the last inequality that $t_{k+1}/t_k = 1 + \frac{1}{13\sqrt{\nu}}$ and $\nu \geq 1$.

Thus using (5.11) one obtains

$$c^\top x_k - \min_{x \in \mathcal{X}} c^\top x \leq \frac{\nu + \sqrt{\nu}/3 + 1/12}{t_k} \leq \frac{2\nu}{t_k}.$$

Observe that $t_k = \left(1 + \frac{1}{13\sqrt{\nu}}\right)^k t_0$, which finally yields

$$c^\top x_k - \min_{x \in \mathcal{X}} c^\top x \leq \frac{2\nu}{t_0}\left(1 + \frac{1}{13\sqrt{\nu}}\right)^{-k}.$$

$\square$

At this point we still need to explain how one can get close to an intial point $x^*(t_0)$ of the central path. This can be done with the following rather clever trick. Assume that one has some point $y_0 \in \mathcal{X}$. The observation is that y_0 is on the central path at $t = 1$ for the problem where c is replaced by $-\nabla F(y_0)$. Now instead of following this central path as $t \to +\infty$, one follows it as $t \to 0$. Indeed for t small enough the central paths for c and for $-\nabla F(y_0)$ will be very close. Thus we iterate the following equations, starting with $t'_0 = 1$,

$$t'_{k+1} = \left(1 - \frac{1}{13\sqrt{\nu}}\right)t'_k,$$
$$y_{k+1} = y_k - [\nabla^2 F(y_k)]^{-1}(-t'_{k+1}\nabla F(y_0) + \nabla F(y_k)).$$

A straightforward analysis shows that for $k = O(\sqrt{\nu}\log\nu)$, which corresponds to $t'_k = 1/\nu^{O(1)}$, one obtains a point y_k such that $\lambda_{F_{t'_k}}(y_k) \leq 1/4$. In other words one can initialize the path-following scheme with $t_0 = t'_k$ and $x_0 = y_k$.

5.3.6 IPMs for LPs and SDPs

We have seen that, roughly, the complexity of interior point methods with a ν-self-concordant barrier is $O\left(M\sqrt{\nu}\log\frac{\nu}{\varepsilon}\right)$, where M is the complexity of computing a Newton direction (which can be done by computing and inverting the Hessian of the barrier). Thus the efficiency of the method is directly related to the *form* of the self-concordant barrier that one can construct for $\mathcal{X}$. It turns out that for LPs and SDPs

one has particularly nice self-concordant barriers. Indeed one can show that $F(x) = -\sum_{i=1}^{n} \log x_i$ is an n-self-concordant barrier on $\mathbb{R}_+^n$, and $F(x) = -\log \det(X)$ is an n-self-concordant barrier on $\mathbb{S}_+^n$. See also Lee and Sidford [2013] for a recent improvement of the basic logarithmic barrier for LPs.

There is one important issue that we overlooked so far. In most interesting cases LPs and SDPs come with *equality constraints*, resulting in a set of constraints $\mathcal{X}$ with empty interior. From a theoretical point of view there is an easy fix, which is to reparametrize the problem as to enforce the variables to live in the subspace spanned by $\mathcal{X}$. This modification also has algorithmic consequences, as the evaluation of the Newton direction will now be different. In fact, rather than doing a reparametrization, one can simply search for Newton directions such that the updated point will stay in $\mathcal{X}$. In other words one has now to solve a convex quadratic optimization problem under linear equality constraints. Luckily using Lagrange multipliers one can find a closed form solution to this problem, and we refer to previous references for more details.

6

Convex optimization and randomness

In this chapter we explore the interplay between optimization and randomness. A key insight, going back to Robbins and Monro [1951], is that first order methods are quite robust: the gradients do not have to be computed exactly to ensure progress towards the optimum. Indeed since these methods usually do many small steps, as long as the gradients are correct *on average*, the error introduced by the gradient approximations will eventually vanish. As we will see below this intuition is correct for non-smooth optimization (since the steps are indeed small) but the picture is more subtle in the case of smooth optimization (recall from Chapter 3 that in this case we take long steps).

We introduce now the main object of this chapter: a (first order) *stochastic* oracle for a convex function $f : \mathcal{X} \to \mathbb{R}$ takes as input a point $x \in \mathcal{X}$ and outputs a random variable $\widetilde{g}(x)$ such that $\mathbb{E}\, \widetilde{g}(x) \in \partial f(x)$. In the case where the query point x is a random variable (possibly obtained from previous queries to the oracle), one assumes that $\mathbb{E}\,(\widetilde{g}(x)|x) \in \partial f(x)$.

The unbiasedness assumption by itself is not enough to obtain rates of convergence, one also needs to make assumptions about the fluctuations of $\widetilde{g}(x)$. Essentially in the non-smooth case we will assume

that there exists $B > 0$ such that $\mathbb{E}\|\widetilde{g}(x)\|_*^2 \leq B^2$ for all $x \in \mathcal{X}$, while in the smooth case we assume that there exists $\sigma > 0$ such that $\mathbb{E}\|\widetilde{g}(x) - \nabla f(x)\|_*^2 \leq \sigma^2$ for all $x \in \mathcal{X}$.

We also note that the situation with a *biased* oracle is quite different, and we refer to d'Aspremont [2008], Schmidt et al. [2011] for some works in this direction.

The two canonical examples of a stochastic oracle in machine learning are as follows.

Let $f(x) = \mathbb{E}_\xi \ell(x, \xi)$ where $\ell(x, \xi)$ should be interpreted as the loss of predictor x on the example ξ. We assume that $\ell(\cdot, \xi)$ is a (differentiable[1]) convex function for any ξ. The goal is to find a predictor with minimal expected loss, that is to minimize f. When queried at x the stochastic oracle can draw ξ from the unknown distribution and report $\nabla_x \ell(x, \xi)$. One obviously has $\mathbb{E}_\xi \nabla_x \ell(x, \xi) \in \partial f(x)$.

The second example is the one described in Section 1.1, where one wants to minimize $f(x) = \frac{1}{m} \sum_{i=1}^{m} f_i(x)$. In this situation a stochastic oracle can be obtained by selecting uniformly at random $I \in [m]$ and reporting $\nabla f_I(x)$.

Observe that the stochastic oracles in the two above cases are quite different. Consider the standard situation where one has access to a data set of i.i.d. samples $\xi_1, \ldots, \xi_m$. Thus in the first case, where one wants to minimize the *expected loss*, one is limited to m queries to the oracle, that is to a *single pass* over the data (indeed one cannot ensure that the conditional expectations are correct if one uses twice a data point). On the contrary for the *empirical loss* where $f_i(x) = \ell(x, \xi_i)$ one can do as many passes as one wishes.

6.1   Non-smooth stochastic optimization

We initiate our study with stochastic mirror descent (S-MD) which is defined as follows: $x_1 \in \operatorname{argmin}_{\mathcal{X} \cap \mathcal{D}} \Phi(x)$, and

$$x_{t+1} = \operatorname*{argmin}_{x \in \mathcal{X} \cap \mathcal{D}} \eta \widetilde{g}(x_t)^\top x + D_\Phi(x, x_t).$$

---

[1]We assume differentiability only for sake of notation here.

In this case equation (4.10) rewrites

$$\sum_{s=1}^{t} \widetilde{g}(x_s)^\top (x_s - x) \leq \frac{R^2}{\eta} + \frac{\eta}{2\rho} \sum_{s=1}^{t} \|\widetilde{g}(x_s)\|_*^2.$$

This immediately yields a rate of convergence thanks to the following simple observation based on the tower rule:

$$
\begin{aligned}
\mathbb{E}f\left(\frac{1}{t}\sum_{s=1}^{t} x_s\right) - f(x) &\leq \frac{1}{t}\mathbb{E}\sum_{s=1}^{t}(f(x_s) - f(x)) \\
&\leq \frac{1}{t}\mathbb{E}\sum_{s=1}^{t}\mathbb{E}(\widetilde{g}(x_s)|x_s)^\top(x_s - x) \\
&= \frac{1}{t}\mathbb{E}\sum_{s=1}^{t}\widetilde{g}(x_s)^\top(x_s - x).
\end{aligned}
$$

We just proved the following theorem.

Theorem 6.1. Let Φ be a mirror map 1-strongly convex on $\mathcal{X} \cap \mathcal{D}$ with respect to $\|\cdot\|$, and let $R^2 = \sup_{x \in \mathcal{X} \cap \mathcal{D}} \Phi(x) - \Phi(x_1)$. Let f be convex. Furthermore assume that the stochastic oracle is such that $\mathbb{E}\|\widetilde{g}(x)\|_*^2 \leq B^2$. Then S-MD with $\eta = \frac{R}{B}\sqrt{\frac{2}{t}}$ satisfies

$$\mathbb{E}f\left(\frac{1}{t}\sum_{s=1}^{t} x_s\right) - \min_{x \in \mathcal{X}} f(x) \leq RB\sqrt{\frac{2}{t}}.$$

Similarly, in the Euclidean and strongly convex case, one can directly generalize Theorem 3.9. Precisely we consider stochastic gradient descent (SGD), that is S-MD with $\Phi(x) = \frac{1}{2}\|x\|_2^2$, with time-varying step size $(\eta_t)_{t \geq 1}$, that is

$$x_{t+1} = \Pi_{\mathcal{X}}(x_t - \eta_t \widetilde{g}(x_t)).$$

Theorem 6.2. Let f be α-strongly convex, and assume that the stochastic oracle is such that $\mathbb{E}\|\widetilde{g}(x)\|_*^2 \leq B^2$. Then SGD with $\eta_s = \frac{2}{\alpha(s+1)}$ satisfies

$$f\left(\sum_{s=1}^{t} \frac{2s}{t(t+1)} x_s\right) - f(x^*) \leq \frac{2B^2}{\alpha(t+1)}.$$

6.2 Smooth stochastic optimization and mini-batch SGD

In the previous section we showed that, for non-smooth optimization, there is basically no cost for having a stochastic oracle instead of an exact oracle. Unfortunately one can show (see e.g. Tsybakov [2003]) that smoothness does not bring any acceleration for a general stochastic oracle[2]. This is in sharp contrast with the exact oracle case where we showed that gradient descent attains a $1/t$ rate (instead of $1/\sqrt{t}$ for non-smooth), and this could even be improved to $1/t^2$ thanks to Nesterov's accelerated gradient descent.

The next result interpolates between the $1/\sqrt{t}$ for stochastic smooth optimization, and the $1/t$ for deterministic smooth optimization. We will use it to propose a useful modification of SGD in the smooth case. The proof is extracted from Dekel et al. [2012].

Theorem 6.3. Let Φ be a mirror map 1-strongly convex on $\mathcal{X} \cap \mathcal{D}$ w.r.t. $\| \cdot \|$, and let $R^2 = \sup_{x \in \mathcal{X} \cap \mathcal{D}} \Phi(x) - \Phi(x_1)$. Let f be convex and β-smooth w.r.t. $\| \cdot \|$. Furthermore assume that the stochastic oracle is such that $\mathbb{E}\|\nabla f(x) - \widetilde{g}(x)\|_*^2 \leq \sigma^2$. Then S-MD with stepsize $\frac{1}{\beta + 1/\eta}$ and $\eta = \frac{R}{\sigma}\sqrt{\frac{2}{t}}$ satisfies

$$\mathbb{E}f\left(\frac{1}{t}\sum_{s=1}^{t} x_{s+1}\right) - f(x^*) \leq R\sigma\sqrt{\frac{2}{t}} + \frac{\beta R^2}{t}.$$

Proof. Using β-smoothness, Cauchy-Schwarz (with $2ab \leq xa^2 + b^2/x$

---

[2]While being true in general this statement does not say anything about specific functions/oracles. For example it was shown in Bach and Moulines [2013] that acceleration can be obtained for the square loss and the logistic loss.

for any $x > 0$), and the 1-strong convexity of Φ, one obtains

$$f(x_{s+1}) - f(x_s)$$

$$\leq \nabla f(x_s)^\top (x_{s+1} - x_s) + \frac{\beta}{2} \|x_{s+1} - x_s\|^2$$

$$= \tilde{g}_s^\top (x_{s+1} - x_s) + (\nabla f(x_s) - \tilde{g}_s)^\top (x_{s+1} - x_s) + \frac{\beta}{2} \|x_{s+1} - x_s\|^2$$

$$\leq \tilde{g}_s^\top (x_{s+1} - x_s) + \frac{\eta}{2} \|\nabla f(x_s) - \tilde{g}_s\|_*^2 + \frac{1}{2}(\beta + 1/\eta)\|x_{s+1} - x_s\|^2$$

$$\leq \tilde{g}_s^\top (x_{s+1} - x_s) + \frac{\eta}{2} \|\nabla f(x_s) - \tilde{g}_s\|_*^2 + (\beta + 1/\eta)D_\Phi(x_{s+1}, x_s).$$

Observe that, using the same argument as to derive (4.9), one has

$$\frac{1}{\beta + 1/\eta} \tilde{g}_s^\top (x_{s+1} - x^*) \leq D_\Phi(x^*, x_s) - D_\Phi(x^*, x_{s+1}) - D_\Phi(x_{s+1}, x_s).$$

Thus

$$f(x_{s+1})$$

$$\leq f(x_s) + \tilde{g}_s^\top (x^* - x_s) + (\beta + 1/\eta)\left(D_\Phi(x^*, x_s) - D_\Phi(x^*, x_{s+1})\right)$$

$$\quad + \frac{\eta}{2} \|\nabla f(x_s) - \tilde{g}_s\|_*^2$$

$$\leq f(x^*) + (\tilde{g}_s - \nabla f(x_s))^\top (x^* - x_s)$$

$$\quad + (\beta + 1/\eta)\left(D_\Phi(x^*, x_s) - D_\Phi(x^*, x_{s+1})\right) + \frac{\eta}{2} \|\nabla f(x_s) - \tilde{g}_s\|_*^2.$$

In particular this yields

$$\mathbb{E}f(x_{s+1}) - f(x^*) \leq (\beta + 1/\eta)\mathbb{E}\left(D_\Phi(x^*, x_s) - D_\Phi(x^*, x_{s+1})\right) + \frac{\eta\sigma^2}{2}.$$

By summing this inequality from $s = 1$ to $s = t$ one can easily conclude with the standard argument. $\square$

We can now propose the following modification of SGD based on the idea of *mini-batches*. Let $m \in \mathbb{N}$, then mini-batch SGD iterates the following equation:

$$x_{t+1} = \Pi_{\mathcal{X}}\left(x_t - \frac{\eta}{m}\sum_{i=1}^m \tilde{g}_i(x_t)\right).$$

where $\widetilde{g}_i(x_t), i = 1, \ldots, m$ are independent random variables (conditionally on x_t) obtained from repeated queries to the stochastic oracle. Assuming that f is β-smooth and that the stochastic oracle is such that $\|\widetilde{g}(x)\|_2 \leq B$, one can obtain a rate of convergence for mini-batch SGD with Theorem 6.3. Indeed one can apply this result with the modified stochastic oracle that returns $\frac{1}{m}\sum_{i=1}^{m}\widetilde{g}_i(x)$, it satisfies

$$\mathbb{E}\|\frac{1}{m}\sum_{i=1}^{m}\widetilde{g}_i(x) - \nabla f(x)\|_2^2 = \frac{1}{m}\mathbb{E}\|\widetilde{g}_1(x) - \nabla f(x)\|_2^2 \leq \frac{2B^2}{m}.$$

Thus one obtains that with t calls to the (original) stochastic oracle, that is t/m iterations of the mini-batch SGD, one has a suboptimality gap bounded by

$$R\sqrt{\frac{2B^2}{m}}\sqrt{\frac{2}{t/m}} + \frac{\beta R^2}{t/m} = 2\frac{RB}{\sqrt{t}} + \frac{m\beta R^2}{t}.$$

Thus as long as $m \leq \frac{B}{R\beta}\sqrt{t}$ one obtains, with mini-batch SGD and t calls to the oracle, a point which is $3\frac{RB}{\sqrt{t}}$-optimal.

Mini-batch SGD can be a better option than basic SGD in at least two situations: (i) When the computation for an iteration of mini-batch SGD can be distributed between multiple processors. Indeed a central unit can send the message to the processors that estimates of the gradient at point x_s have to be computed, then each processor can work independently and send back the estimate they obtained. (ii) Even in a serial setting mini-batch SGD can sometimes be advantageous, in particular if some calculations can be re-used to compute several estimated gradients at the same point.

6.3   Sum of smooth and strongly convex functions

Let us examine in more details the main example from Section 1.1. That is one is interested in the unconstrained minimization of

$$f(x) = \frac{1}{m}\sum_{i=1}^{m}f_i(x),$$

where $f_1, \ldots, f_m$ are β-smooth and convex functions, and f is α-strongly convex. Typically in machine learning α can be as small as

$1/m$, while β is of order of a constant. In other words the condition number $\kappa = \beta/\alpha$ can be as large as $\Omega(m)$. Let us now compare the basic gradient descent, that is

$$x_{t+1} = x_t - \frac{\eta}{m} \sum_{i=1}^{m} \nabla f_i(x),$$

to SGD

$$x_{t+1} = x_t - \eta \nabla f_{i_t}(x),$$

where i_t is drawn uniformly at random in $[m]$ (independently of everything else). Theorem 3.10 shows that gradient descent requires $O(m\kappa \log(1/\varepsilon))$ gradient computations (which can be improved to $O(m\sqrt{\kappa} \log(1/\varepsilon))$ with Nesterov's accelerated gradient descent), while Theorem 6.2 shows that SGD (with appropriate averaging) requires $O(1/(\alpha\varepsilon))$ gradient computations. Thus one can obtain a low accuracy solution reasonably fast with SGD, but for high accuracy the basic gradient descent is more suitable. Can we get the best of both worlds? This question was answered positively in Le Roux et al. [2012] with SAG (Stochastic Averaged Gradient) and in Shalev-Shwartz and Zhang [2013a] with SDCA (Stochastic Dual Coordinate Ascent). These methods require only $O((m + \kappa) \log(1/\varepsilon))$ gradient computations. We describe below the SVRG (Stochastic Variance Reduced Gradient descent) algorithm from Johnson and Zhang [2013] which makes the main ideas of SAG and SDCA more transparent (see also Defazio et al. [2014] for more on the relation between these different methods). We also observe that a natural question is whether one can obtain a Nesterov's accelerated version of these algorithms that would need only $O((m+\sqrt{m\kappa}) \log(1/\varepsilon))$, see Shalev-Shwartz and Zhang [2013b], Zhang and Xiao [2014], Agarwal and Bottou [2014] for recent works on this question.

To obtain a linear rate of convergence one needs to make "big steps", that is the step-size should be of order of a constant. In SGD the step-size is typically of order $1/\sqrt{t}$ because of the variance introduced by the stochastic oracle. The idea of SVRG is to "center" the output of the stochastic oracle in order to reduce the variance. Precisely instead of feeding $\nabla f_i(x)$ into the gradient descent one would use $\nabla f_i(x) -$

$\nabla f_i(y) + \nabla f(y)$ where y is a centering sequence. This is a sensible idea since, when x and y are close to the optimum, one should have that $\nabla f_i(x) - \nabla f_i(y)$ will have a small variance, and of course $\nabla f(y)$ will also be small (note that $\nabla f_i(x)$ by itself is not necessarily small). This intuition is made formal with the following lemma.

Lemma 6.4. Let $f_1, \ldots f_m$ be β-smooth convex functions on $\mathbb{R}^n$, and i be a random variable uniformly distributed in $[m]$. Then

$$\mathbb{E}\|\nabla f_i(x) - \nabla f_i(x^*)\|_2^2 \leq 2\beta(f(x) - f(x^*)).$$

Proof. Let $g_i(x) = f_i(x) - f_i(x^*) - \nabla f_i(x^*)^\top (x - x^*)$. By convexity of f_i one has $g_i(x) \geq 0$ for any x and in particular using (3.5) this yields $-g_i(x) \leq -\frac{1}{2\beta}\|\nabla g_i(x)\|_2^2$ which can be equivalently written as

$$\|\nabla f_i(x) - \nabla f_i(x^*)\|_2^2 \leq 2\beta(f_i(x) - f_i(x^*) - \nabla f_i(x^*)^\top (x - x^*)).$$

Taking expectation with respect to i and observing that $\mathbb{E}\nabla f_i(x^*) = \nabla f(x^*) = 0$ yields the claimed bound. $\qquad\square$

On the other hand the computation of $\nabla f(y)$ is expensive (it requires m gradient computations), and thus the centering sequence should be updated more rarely than the main sequence. These ideas lead to the following epoch-based algorithm.

Let $y^{(1)} \in \mathbb{R}^n$ be an arbitrary initial point. For $s = 1, 2 \ldots$, let $x_1^{(s)} = y^{(s)}$. For $t = 1, \ldots, k$ let

$$x_{t+1}^{(s)} = x_t^{(s)} - \eta\left(\nabla f_{i_t^{(s)}}(x_t^{(s)}) - \nabla f_{i_t^{(s)}}(y^{(s)}) + \nabla f(y^{(s)})\right),$$

where $i_t^{(s)}$ is drawn uniformly at random (and independently of everything else) in $[m]$. Also let

$$y^{(s+1)} = \frac{1}{k}\sum_{t=1}^{k} x_t^{(s)}.$$

Theorem 6.5. Let $f_1, \ldots f_m$ be β-smooth convex functions on $\mathbb{R}^n$ and f be α-strongly convex. Then SVRG with $\eta = \frac{1}{10\beta}$ and $k = 20\kappa$ satisfies

$$\mathbb{E}f(y^{(s+1)}) - f(x^*) \leq 0.9^s(f(y^{(1)}) - f(x^*)).$$

Proof. We fix a phase $s \geq 1$ and we denote by $\mathbb{E}$ the expectation taken with respect to $i_1^{(s)}, \ldots, i_k^{(s)}$. We show below that

$$\mathbb{E}f(y^{(s+1)}) - f(x^*) = \mathbb{E}f\left(\frac{1}{k}\sum_{t=1}^{k} x_t^{(s)}\right) - f(x^*) \leq 0.9(f(y^{(s)}) - f(x^*)),$$

which clearly implies the theorem. To simplify the notation in the following we drop the dependency on s, that is we want to show that

$$\mathbb{E}f\left(\frac{1}{k}\sum_{t=1}^{k} x_t\right) - f(x^*) \leq 0.9(f(y) - f(x^*)). \tag{6.1}$$

We start as for the proof of Theorem 3.10 (analysis of gradient descent for smooth and strongly convex functions) with

$$\|x_{t+1} - x^*\|_2^2 = \|x_t - x^*\|_2^2 - 2\eta v_t^\top (x_t - x^*) + \eta^2 \|v_t\|_2^2, \tag{6.2}$$

where

$$v_t = \nabla f_{i_t}(x_t) - \nabla f_{i_t}(y) + \nabla f(y).$$

Using Lemma 6.4, we upper bound $\mathbb{E}_{i_t}\|v_t\|_2^2$ as follows (also recall that $\mathbb{E}\|X - \mathbb{E}(X)\|_2^2 \leq \mathbb{E}\|X\|_2^2$, and $\mathbb{E}_{i_t}\nabla f_{i_t}(x^*) = 0$):

$$\mathbb{E}_{i_t}\|v_t\|_2^2$$
$$\leq 2\mathbb{E}_{i_t}\|\nabla f_{i_t}(x_t) - \nabla f_{i_t}(x^*)\|_2^2 + 2\mathbb{E}_{i_t}\|\nabla f_{i_t}(y) - \nabla f_{i_t}(x^*) - \nabla f(y)\|_2^2$$
$$\leq 2\mathbb{E}_{i_t}\|\nabla f_{i_t}(x_t) - \nabla f_{i_t}(x^*)\|_2^2 + 2\mathbb{E}_{i_t}\|\nabla f_{i_t}(y) - \nabla f_{i_t}(x^*)\|_2^2$$
$$\leq 4\beta(f(x_t) - f(x^*) + f(y) - f(x^*)). \tag{6.3}$$

Also observe that

$$\mathbb{E}_{i_t}v_t^\top (x_t - x^*) = \nabla f(x_t)^\top (x_t - x^*) \geq f(x_t) - f(x^*),$$

and thus plugging this into (6.2) together with (6.3) one obtains

$$\mathbb{E}_{i_t}\|x_{t+1} - x^*\|_2^2 \leq \|x_t - x^*\|_2^2 - 2\eta(1 - 2\beta\eta)(f(x_t) - f(x^*))$$
$$+ 4\beta\eta^2(f(y) - f(x^*)).$$

Summing the above inequality over $t = 1, \ldots, k$ yields

$$\mathbb{E}\|x_{k+1} - x^*\|_2^2 \leq \|x_1 - x^*\|_2^2 - 2\eta(1 - 2\beta\eta)\mathbb{E}\sum_{t=1}^{k}(f(x_t) - f(x^*))$$
$$+ 4\beta\eta^2 k(f(y) - f(x^*)).$$

Noting that $x_1 = y$ and that by α-strong convexity one has $f(x) - f(x^*) \geq \frac{\alpha}{2}\|x - x^*\|_2^2$, one can rearrange the above display to obtain

$$\mathbb{E}f\left(\frac{1}{k}\sum_{t=1}^{k} x_t\right) - f(x^*) \leq \left(\frac{1}{\alpha\eta(1-2\beta\eta)k} + \frac{2\beta\eta}{1-2\beta\eta}\right)(f(y) - f(x^*)).$$

Using that $\eta = \frac{1}{10\beta}$ and $k = 20\kappa$ finally yields (6.1) which itself concludes the proof. $\square$

6.4   Random coordinate descent

We assume throughout this section that f is a convex and differentiable function on $\mathbb{R}^n$, with a unique[3] minimizer x^*. We investigate one of the simplest possible scheme to optimize f, the random coordinate descent (RCD) method. In the following we denote $\nabla_i f(x) = \frac{\partial f}{\partial x_i}(x)$. RCD is defined as follows, with an arbitrary initial point $x_1 \in \mathbb{R}^n$,

$$x_{s+1} = x_s - \eta\nabla_{i_s} f(x)e_{i_s},$$

where i_s is drawn uniformly at random from $[n]$ (and independently of everything else).

   One can view RCD as SGD with the specific oracle $\widetilde{g}(x) = n\nabla_I f(x)e_I$ where I is drawn uniformly at random from $[n]$. Clearly $\mathbb{E}\widetilde{g}(x) = \nabla f(x)$, and furthermore

$$\mathbb{E}\|\widetilde{g}(x)\|_2^2 = \frac{1}{n}\sum_{i=1}^{n} \|n\nabla_i f(x)e_i\|_2^2 = n\|\nabla f(x)\|_2^2.$$

Thus using Theorem 6.1 (with $\Phi(x) = \frac{1}{2}\|x\|_2^2$, that is S-MD being SGD) one immediately obtains the following result.

Theorem 6.6. Let f be convex and L-Lipschitz on $\mathbb{R}^n$, then RCD with $\eta = \frac{R}{L}\sqrt{\frac{2}{nt}}$ satisfies

$$\mathbb{E}f\left(\frac{1}{t}\sum_{s=1}^{t} x_s\right) - \min_{x\in\mathcal{X}} f(x) \leq RL\sqrt{\frac{2n}{t}}.$$

---

[3]Uniqueness is only assumed for sake of notation.

Somewhat unsurprisingly RCD requires n times more iterations than gradient descent to obtain the same accuracy. In the next section, we will see that this statement can be greatly improved by taking into account directional smoothness.

6.4.1 RCD for coordinate-smooth optimization

We assume now directional smoothness for f, that is there exists $\beta_1, \ldots, \beta_n$ such that for any $i \in [n], x \in \mathbb{R}^n$ and $u \in \mathbb{R}$,

$$|\nabla_i f(x + ue_i) - \nabla_i f(x)| \leq \beta_i |u|.$$

If f is twice differentiable then this is equivalent to $(\nabla^2 f(x))_{i,i} \leq \beta_i$. In particular, since the maximal eigenvalue of a matrix is upper bounded by its trace, one can see that the directional smoothness implies that f is β-smooth with $\beta \leq \sum_{i=1}^n \beta_i$. We now study the following "aggressive" RCD, where the step-sizes are of order of the inverse smoothness:

$$x_{s+1} = x_s - \frac{1}{\beta_{i_s}} \nabla_{i_s} f(x) e_{i_s}.$$

Furthermore we study a more general sampling distribution than uniform, precisely for $\gamma \geq 0$ we assume that i_s is drawn (independently) from the distribution p_γ defined by

$$p_\gamma(i) = \frac{\beta_i^\gamma}{\sum_{j=1}^n \beta_j^\gamma}, i \in [n].$$

This algorithm was proposed in Nesterov [2012], and we denote it by RCD(γ). Observe that, up to a preprocessing step of complexity $O(n)$, one can sample from p_γ in time $O(\log(n))$.

The following rate of convergence is derived in Nesterov [2012], using the dual norms $\|\cdot\|_{[\gamma]}, \|\cdot\|_{[\gamma]}^*$ defined by

$$\|x\|_{[\gamma]} = \sqrt{\sum_{i=1}^n \beta_i^\gamma x_i^2}, \text{ and } \|x\|_{[\gamma]}^* = \sqrt{\sum_{i=1}^n \frac{1}{\beta_i^\gamma} x_i^2}.$$

Theorem 6.7. Let f be convex and such that $u \in \mathbb{R} \mapsto f(x + ue_i)$ is β_i-smooth for any $i \in [n], x \in \mathbb{R}^n$. Then RCD($\gamma$) satisfies for $t \geq 2$,

$$\mathbb{E} f(x_t) - f(x^*) \leq \frac{2R_{1-\gamma}^2(x_1) \sum_{i=1}^n \beta_i^\gamma}{t-1},$$

where
$$R_{1-\gamma}(x_1) = \sup_{x \in \mathbb{R}^n : f(x) \leq f(x_1)} \|x - x^*\|_{[1-\gamma]}.$$

Recall from Theorem 3.3 that in this context the basic gradient descent attains a rate of $\beta \|x_1 - x^*\|_2^2 / t$ where $\beta \leq \sum_{i=1}^n \beta_i$ (see the discussion above). Thus we see that RCD(1) greatly improves upon gradient descent for functions where β is of order of $\sum_{i=1}^n \beta_i$. Indeed in this case both methods attain the same accuracy after a fixed number of iterations, but the iterations of coordinate descent are potentially much cheaper than the iterations of gradient descent.

Proof. By applying (3.5) to the β_i-smooth function $u \in \mathbb{R} \mapsto f(x + ue_i)$ one obtains
$$f\left(x - \frac{1}{\beta_i} \nabla_i f(x) e_i\right) - f(x) \leq -\frac{1}{2\beta_i} (\nabla_i f(x))^2.$$

We use this as follows:
$$
\begin{aligned}
\mathbb{E}_{i_s} f(x_{s+1}) - f(x_s) &= \sum_{i=1}^n p_\gamma(i) \left(f\left(x_s - \frac{1}{\beta_i} \nabla_i f(x_s) e_i\right) - f(x_s) \right) \\
&\leq -\sum_{i=1}^n \frac{p_\gamma(i)}{2\beta_i} (\nabla_i f(x_s))^2 \\
&= -\frac{1}{2 \sum_{i=1}^n \beta_i^\gamma} \left(\|\nabla f(x_s)\|_{[1-\gamma]}^* \right)^2.
\end{aligned}
$$

Denote $\delta_s = \mathbb{E} f(x_s) - f(x^*)$. Observe that the above calculation can be used to show that $f(x_{s+1}) \leq f(x_s)$ and thus one has, by definition of $R_{1-\gamma}(x_1)$,
$$
\begin{aligned}
\delta_s &\leq \nabla f(x_s)^\top (x_s - x^*) \\
&\leq \|x_s - x^*\|_{[1-\gamma]} \|\nabla f(x_s)\|_{[1-\gamma]}^* \\
&\leq R_{1-\gamma}(x_1) \|\nabla f(x_s)\|_{[1-\gamma]}^*.
\end{aligned}
$$

Thus putting together the above calculations one obtains
$$\delta_{s+1} \leq \delta_s - \frac{1}{2R_{1-\gamma}^2(x_1) \sum_{i=1}^n \beta_i^\gamma} \delta_s^2.$$

The proof can be concluded with similar computations than for Theorem 3.3. $\qquad\square$

We discussed above the specific case of $\gamma = 1$. Both $\gamma = 0$ and $\gamma = 1/2$ also have an interesting behavior, and we refer to Nesterov [2012] for more details. The latter paper also contains a discussion of high probability results and potential acceleration à la Nesterov. We also refer to Richtárik and Takác [2012] for a discussion of RCD in a distributed setting.

6.4.2 RCD for smooth and strongly convex optimization

If in addition to directional smoothness one also assumes strong convexity, then RCD attains in fact a linear rate.

Theorem 6.8. Let $\gamma \geq 0$. Let f be α-strongly convex w.r.t. $\|\cdot\|_{[1-\gamma]}$, and such that $u \in \mathbb{R} \mapsto f(x + ue_i)$ is β_i-smooth for any $i \in [n], x \in \mathbb{R}^n$. Let $\kappa_\gamma = \frac{\sum_{i=1}^n \beta_i^\gamma}{\alpha}$, then $\text{RCD}(\gamma)$ satisfies

$$\mathbb{E}f(x_{t+1}) - f(x^*) \leq \left(1 - \frac{1}{\kappa_\gamma}\right)^t (f(x_1) - f(x^*)).$$

We use the following elementary lemma.

Lemma 6.9. Let f be α-strongly convex w.r.t. $\|\cdot\|$ on $\mathbb{R}^n$, then

$$f(x) - f(x^*) \leq \frac{1}{2\alpha}\|\nabla f(x)\|_*^2.$$

Proof. By strong convexity, Hölder's inequality, and an elementary calculation,

$$
\begin{aligned}
f(x) - f(y) &\leq \nabla f(x)^\top (x - y) - \frac{\alpha}{2}\|x - y\|_2^2 \\
&\leq \|\nabla f(x)\|_* \|x - y\| - \frac{\alpha}{2}\|x - y\|_2^2 \\
&\leq \frac{1}{2\alpha}\|\nabla f(x)\|_*^2,
\end{aligned}
$$

which concludes the proof by taking $y = x^*$. $\qquad\square$

We can now prove Theorem 6.8.

Proof. In the proof of Theorem 6.7 we showed that

$$\delta_{s+1} \leq \delta_s - \frac{1}{2\sum_{i=1}^n \beta_i^\gamma}\left(\|\nabla f(x_s)\|_{[1-\gamma]}^*\right)^2.$$

On the other hand Lemma 6.9 shows that

$$\left(\|\nabla f(x_s)\|_{[1-\gamma]}^*\right)^2 \geq 2\alpha\delta_s.$$

The proof is concluded with straightforward calculations. □

6.5 Acceleration by randomization for saddle points

We explore now the use of randomness for saddle point computations. That is we consider the context of Section 5.2.1 with a stochastic oracle of the following form: given $z = (x,y) \in \mathcal{X} \times \mathcal{Y}$ it outputs $\widetilde{g}(z) = (\widetilde{g}_{\mathcal{X}}(x,y), \widetilde{g}_{\mathcal{Y}}(x,y))$ where $\mathbb{E}\left(\widetilde{g}_{\mathcal{X}}(x,y)|x,y\right) \in \partial_x\varphi(x,y)$, and $\mathbb{E}\left(\widetilde{g}_{\mathcal{Y}}(x,y)|x,y\right) \in \partial_y(-\varphi(x,y))$. Instead of using true subgradients as in SP-MD (see Section 5.2.2) we use here the outputs of the stochastic oracle. We refer to the resulting algorithm as S-SP-MD (Stochastic Saddle Point Mirror Descent). Using the same reasoning than in Section 6.1 and Section 5.2.2 one can derive the following theorem.

Theorem 6.10. Assume that the stochastic oracle is such that $\mathbb{E}\left(\|\widetilde{g}_{\mathcal{X}}(x,y)\|_{\mathcal{X}}^*\right)^2 \leq B_{\mathcal{X}}^2$, and $\mathbb{E}\left(\|\widetilde{g}_{\mathcal{Y}}(x,y)\|_{\mathcal{Y}}^*\right)^2 \leq B_{\mathcal{Y}}^2$. Then S-SP-MD with $a = \frac{B_{\mathcal{X}}}{R_{\mathcal{X}}}$, $b = \frac{B_{\mathcal{Y}}}{R_{\mathcal{Y}}}$, and $\eta = \sqrt{\frac{2}{t}}$ satisfies

$$\mathbb{E}\left(\max_{y\in\mathcal{Y}}\varphi\left(\frac{1}{t}\sum_{s=1}^{t}x_s, y\right) - \min_{x\in\mathcal{X}}\varphi\left(x, \frac{1}{t}\sum_{s=1}^{t}y_s\right)\right) \leq (R_{\mathcal{X}}B_{\mathcal{X}}+R_{\mathcal{Y}}B_{\mathcal{Y}})\sqrt{\frac{2}{t}}.$$

Using S-SP-MD we revisit the examples of Section 5.2.4 and Section 5.2.4. In both cases one has $\varphi(x,y) = x^\top A y$ (with A_i being the ith column of A), and thus $\nabla_x\varphi(x,y) = Ay$ and $\nabla_y\varphi(x,y) = A^\top x$.

Matrix games. Here $x \in \Delta_n$ and $y \in \Delta_m$. Thus there is a quite natural stochastic oracle:

$$\widetilde{g}_{\mathcal{X}}(x,y) = A_I, \text{ where } I \in [m] \text{ is drawn according to } y \in \Delta_m, \quad (6.4)$$

and $\forall i \in [m]$,

$$\widetilde{g}_{\mathcal{Y}}(x,y)(i) = A_i(J), \text{ where } J \in [n] \text{ is drawn according to } x \in \Delta_n. \quad (6.5)$$

Clearly $\|\widetilde{g}_\mathcal{X}(x,y)\|_\infty \leq \|A\|_{\max}$ and $\|\widetilde{g}_\mathcal{X}(x,y)\|_\infty \leq \|A\|_{\max}$, which implies that S-SP-MD attains an ε-optimal pair of points with $O\left(\|A\|_{\max}^2 \log(n+m)/\varepsilon^2\right)$ iterations. Furthermore the computational complexity of a step of S-SP-MD is dominated by drawing the indices I and J which takes $O(n+m)$. Thus overall the complexity of getting an ε-optimal Nash equilibrium with S-SP-MD is $O\left(\|A\|_{\max}^2 (n+m) \log(n+m)/\varepsilon^2\right)$. While the dependency on ε is worse than for SP-MP (see Section 5.2.4), the dependencies on the dimensions is $\widetilde{O}(n+m)$ instead of $\widetilde{O}(nm)$. In particular, quite astonishingly, this is *sublinear* in the size of the matrix A. The possibility of sublinear algorithms for this problem was first observed in Grigoriadis and Khachiyan [1995].

Linear classification. Here $x \in \mathrm{B}_{2,n}$ and $y \in \Delta_m$. Thus the stochastic oracle for the x-subgradient can be taken as in (6.4) but for the y-subgradient we modify (6.5) as follows. For a vector x we denote by x^2 the vector such that $x^2(i) = x(i)^2$. For all $i \in [m]$, $\widetilde{g}_\mathcal{Y}(x,y)(i) = \frac{\|x\|^2}{x(j)} A_i(J)$, where $J \in [n]$ is drawn according to $\frac{x^2}{\|x\|_2^2} \in \Delta_n$. Note that one indeed has $\mathbb{E}(\widetilde{g}_\mathcal{Y}(x,y)(i)|x,y) = \sum_{j=1}^n x(j) A_i(j) = (A^\top x)(i)$. Furthermore $\|\widetilde{g}_\mathcal{X}(x,y)\|_2 \leq B$, and

$$\mathbb{E}(\|\widetilde{g}_\mathcal{Y}(x,y)\|_\infty^2|x,y) = \sum_{j=1}^n \frac{x(j)^2}{\|x\|_2^2} \max_{i \in [m]} \left(\frac{\|x\|^2}{x(j)} A_i(j)\right)^2 \leq \sum_{j=1}^n \max_{i \in [m]} A_i(j)^2.$$

Unfortunately this last term can be $O(n)$. However it turns out that one can do a more careful analysis of mirror descent in terms of local norms, which allows to prove that the "local variance" is dimension-free. We refer to Bubeck and Cesa-Bianchi [2012] for more details on these local norms, and to Clarkson et al. [2012] for the specific details in the linear classification situation.

6.6 Convex relaxation and randomized rounding

In this section we briefly discuss the concept of convex relaxation, and the use of randomization to find approximate solutions. By now there is an enormous literature on these topics, and we refer to Barak [2014]

for further pointers.

We study here the seminal example of MAXCUT. This problem can be described as follows. Let $A \in \mathbb{R}_+^{n \times n}$ be a symmetric matrix of non-negative weights. The entry $A_{i,j}$ is interpreted as a measure of the "dissimilarity" between point i and point j. The goal is to find a partition of $[n]$ into two sets, $S \subset [n]$ and S^c, so as to maximize the total dissimilarity between the two groups: $\sum_{i \in S, j \in S^c} A_{i,j}$. Equivalently MAXCUT corresponds to the following optimization problem:

$$\max_{x \in \{-1,1\}^n} \frac{1}{2} \sum_{i,j=1}^{n} A_{i,j}(x_i - x_j)^2. \tag{6.6}$$

Viewing A as the (weighted) adjacency matrix of a graph, one can rewrite (6.6) as follows, using the graph Laplacian $L = D - A$ where D is the diagonal matrix with entries $(\sum_{j=1}^{n} A_{i,j})_{i \in [n]}$,

$$\max_{x \in \{-1,1\}^n} x^\top L x. \tag{6.7}$$

It turns out that this optimization problem is **NP**-hard, that is the existence of a polynomial time algorithm to solve (6.7) would prove that $\mathbf{P} = \mathbf{NP}$. The combinatorial difficulty of this problem stems from the hypercube constraint. Indeed if one replaces $\{-1,1\}^n$ by the Euclidean sphere, then one obtains an efficiently solvable problem (it is the problem of computing the maximal eigenvalue of L).

We show now that, while (6.7) is a difficult optimization problem, it is in fact possible to find relatively good *approximate* solutions by using the power of randomization. Let ζ be uniformly drawn on the hypercube $\{-1,1\}^n$, then clearly

$$\mathbb{E} \, \zeta^\top L \zeta = \sum_{i,j=1, i \neq j}^{n} A_{i,j} \geq \frac{1}{2} \max_{x \in \{-1,1\}^n} x^\top L x.$$

This means that, on average, ζ is a $1/2$-approximate solution to (6.7). Furthermore it is immediate that the above expectation bound implies that, with probability at least ε, ζ is a $(1/2 - \varepsilon)$-approximate solution. Thus by repeatedly sampling uniformly from the hypercube one can get arbitrarily close (with probability approaching 1) to a $1/2$-approximation of MAXCUT.

Next we show that one can obtain an even better approximation ratio by combining the power of convex optimization and randomization. This approach was pioneered by Goemans and Williamson [1995]. The Goemans-Williamson algorithm is based on the following inequality

$$\max_{x \in \{-1,1\}^n} x^\top L x = \max_{x \in \{-1,1\}^n} \langle L, x x^\top \rangle \leq \max_{X \in \mathbb{S}_+^n, X_{i,i}=1, i \in [n]} \langle L, X \rangle.$$

The right hand side in the above display is known as the *convex (or SDP) relaxation* of MAXCUT. The convex relaxation is an SDP and thus one can find its solution efficiently with Interior Point Methods (see Section 5.3). The following result states both the Goemans-Williamson strategy and the corresponding approximation ratio.

Theorem 6.11. Let Σ be the solution to the SDP relaxation of MAXCUT. Let $\xi \sim \mathcal{N}(0, \Sigma)$ and $\zeta = \text{sign}(\xi) \in \{-1, 1\}^n$. Then

$$\mathbb{E} \, \zeta^\top L \zeta \geq 0.878 \max_{x \in \{-1,1\}^n} x^\top L x.$$

The proof of this result is based on the following elementary geometric lemma.

Lemma 6.12. Let $\xi \sim \mathcal{N}(0, \Sigma)$ with $\Sigma_{i,i} = 1$ for $i \in [n]$, and $\zeta = \text{sign}(\xi)$. Then

$$\mathbb{E} \, \zeta_i \zeta_j = \frac{2}{\pi} \arcsin(\Sigma_{i,j}).$$

Proof. Let $V \in \mathbb{R}^{n \times n}$ (with i^{th} row $V_i^\top$) be such that $\Sigma = VV^\top$. Note that since $\Sigma_{i,i} = 1$ one has $\|V_i\|_2 = 1$ (remark also that necessarily $|\Sigma_{i,j}| \leq 1$, which will be important in the proof of Theorem 6.11). Let $\varepsilon \sim \mathcal{N}(0, \mathrm{I}_n)$ be such that $\xi = V\varepsilon$. Then $\zeta_i = \text{sign}(V_i^\top \varepsilon)$, and in particular

$$
\begin{aligned}
\mathbb{E} \, \zeta_i \zeta_j &= \mathbb{P}(V_i^\top \varepsilon \geq 0 \text{ and } V_j^\top \varepsilon \geq 0) + \mathbb{P}(V_i^\top \varepsilon \leq 0 \text{ and } V_j^\top \varepsilon \leq 0 \\
&\quad - \mathbb{P}(V_i^\top \varepsilon \geq 0 \text{ and } V_j^\top \varepsilon < 0) - \mathbb{P}(V_i^\top \varepsilon < 0 \text{ and } V_j^\top \varepsilon \geq 0) \\
&= 2\mathbb{P}(V_i^\top \varepsilon \geq 0 \text{ and } V_j^\top \varepsilon \geq 0) - 2\mathbb{P}(V_i^\top \varepsilon \geq 0 \text{ and } V_j^\top \varepsilon < 0) \\
&= \mathbb{P}(V_j^\top \varepsilon \geq 0 | V_i^\top \varepsilon \geq 0) - \mathbb{P}(V_j^\top \varepsilon < 0 | V_i^\top \varepsilon \geq 0) \\
&= 1 - 2\mathbb{P}(V_j^\top \varepsilon < 0 | V_i^\top \varepsilon \geq 0).
\end{aligned}
$$

Now a quick picture shows that $\mathbb{P}(V_j^\top \varepsilon < 0 | V_i^\top \varepsilon \geq 0) = \frac{1}{\pi}\arccos(V_i^\top V_j)$ (recall that $\varepsilon/\|\varepsilon\|_2$ is uniform on the Euclidean sphere). Using the fact that $V_i^\top V_j = \Sigma_{i,j}$ and $\arccos(x) = \frac{\pi}{2} - \arcsin(x)$ conclude the proof. $\quad\square$

We can now get to the proof of Theorem 6.11.

Proof. We shall use the following inequality:

$$1 - \frac{2}{\pi}\arcsin(t) \geq 0.878(1 - t), \ \forall t \in [-1, 1]. \tag{6.8}$$

Also remark that for $X \in \mathbb{R}^{n \times n}$ such that $X_{i,i} = 1$, one has

$$\langle L, X \rangle = \sum_{i,j=1}^{n} A_{i,j}(1 - X_{i,j}),$$

and in particular for $x \in \{-1, 1\}^n$, $x^\top L x = \sum_{i,j=1}^{n} A_{i,j}(1 - x_i x_j)$. Thus, using Lemma 6.12, and the facts that $A_{i,j} \geq 0$ and $|\Sigma_{i,j}| \leq 1$ (see the proof of Lemma 6.12), one has

$$
\begin{aligned}
\mathbb{E} \ \zeta^\top L \zeta &= \sum_{i,j=1}^{n} A_{i,j} \left(1 - \frac{2}{\pi}\arcsin\left(\Sigma_{i,j}\right)\right) \\
&\geq 0.878 \sum_{i,j=1}^{n} A_{i,j} \left(1 - \Sigma_{i,j}\right) \\
&= 0.878 \max_{X \in \mathbb{S}_+^n, X_{i,i}=1, i \in [n]} \langle L, X \rangle \\
&\geq 0.878 \max_{x \in \{-1,1\}^n} x^\top L x.
\end{aligned}
$$

$\square$

Theorem 6.11 depends on the form of the Laplacian L (insofar as (6.8) was used). We show next a result from Nesterov [1997] that applies to any positive semi-definite matrix, at the expense of the constant of approximation. Precisely we are now interested in the following optimization problem:

$$\max_{x \in \{-1,1\}^n} x^\top B x. \tag{6.9}$$

The corresponding SDP relaxation is

$$\max_{X \in \mathbb{S}_+^n, X_{i,i}=1, i \in [n]} \langle B, X \rangle.$$

Theorem 6.13. Let Σ be the solution to the SDP relaxation of (6.9). Let $\xi \sim \mathcal{N}(0, \Sigma)$ and $\zeta = \text{sign}(\xi) \in \{-1, 1\}^n$. Then

$$\mathbb{E} \, \zeta^\top B \zeta \geq \frac{2}{\pi} \max_{x \in \{-1,1\}^n} x^\top B x.$$

Proof. Lemma 6.12 shows that

$$\mathbb{E} \, \zeta^\top B \zeta = \sum_{i,j=1}^n B_{i,j} \frac{2}{\pi} \arcsin(X_{i,j}) = \frac{2}{\pi} \langle B, \arcsin(X) \rangle.$$

Thus to prove the result it is enough to show that $\langle B, \arcsin(\Sigma) \rangle \geq \langle B, \Sigma \rangle$, which is itself implied by $\arcsin(\Sigma) \succeq \Sigma$ (the implication is true since B is positive semi-definite, just write the eigendecomposition). Now we prove the latter inequality via a Taylor expansion. Indeed recall that $|\Sigma_{i,j}| \leq 1$ and thus denoting by $A^{\circ \alpha}$ the matrix where the entries are raised to the power α one has

$$\arcsin(\Sigma) = \sum_{k=0}^{+\infty} \frac{\binom{2k}{k}}{4^k (2k+1)} \Sigma^{\circ(2k+1)} = \Sigma + \sum_{k=1}^{+\infty} \frac{\binom{2k}{k}}{4^k (2k+1)} \Sigma^{\circ(2k+1)}.$$

Finally one can conclude using the fact if $A, B \succeq 0$ then $A \circ B \succeq 0$. This can be seen by writing $A = VV^\top$, $B = UU^\top$, and thus

$$(A \circ B)_{i,j} = V_i^\top V_j U_i^\top U_j = \text{Tr}(U_j V_j^\top V_i U_i^\top) = \langle V_i U_i^\top, V_j U_j^\top \rangle.$$

In other words $A \circ B$ is a Gram-matrix and, thus it is positive semi-definite. $\qquad \square$

6.7 Random walk based methods

Randomization naturally suggests itself in the center of gravity method (see Section 2.1), as a way to circumvent the exact calculation of the center of gravity. This idea was proposed and developed in Bertsimas and Vempala [2004]. We give below a condensed version of the main ideas of this paper.

Assuming that one can draw independent points $X_1, \ldots, X_N$ uniformly at random from the current set $\mathcal{S}_t$, one could replace c_t by $\hat{c}_t = \frac{1}{N} \sum_{i=1}^N X_i$. Bertsimas and Vempala [2004] proved the following

generalization of Lemma 2.2 for the situation where one cuts a convex set through a point close the center of gravity. Recall that a convex set $\mathcal{K}$ is in isotropic position if $\mathbb{E}X = 0$ and $\mathbb{E}XX^\top = \mathrm{I}_n$, where X is a random variable drawn uniformly at random from $\mathcal{K}$. Note in particular that this implies $\mathbb{E}\|X\|_2^2 = n$. We also say that $\mathcal{K}$ is in near-isotropic position if $\frac{1}{2}\mathrm{I}_n \preceq \mathbb{E}XX^\top \preceq \frac{3}{2}\mathrm{I}_n$.

Lemma 6.14. Let $\mathcal{K}$ be a convex set in isotropic position. Then for any $w \in \mathbb{R}^n, w \neq 0, z \in \mathbb{R}^n$, one has

$$\mathrm{Vol}\left(\mathcal{K} \cap \{x \in \mathbb{R}^n : (x - z)^\top w \geq 0\}\right) \geq \left(\frac{1}{e} - \|z\|_2\right)\mathrm{Vol}(\mathcal{K}).$$

Thus if one can ensure that $\mathcal{S}_t$ is in (near) isotropic position, and $\|c_t - \hat{c}_t\|_2$ is small (say smaller than 0.1), then the randomized center of gravity method (which replaces c_t by $\hat{c}_t$) will converge at the same speed than the original center of gravity method.

Assuming that $\mathcal{S}_t$ is in isotropic position one immediately obtains $\mathbb{E}\|c_t - \hat{c}_t\|_2^2 = \frac{n}{N}$, and thus by Chebyshev's inequality one has $\mathbb{P}(\|c_t - \hat{c}_t\|_2 > 0.1) \leq 100\frac{n}{N}$. In other words with $N = O(n)$ one can ensure that the randomized center of gravity method makes progress on a constant fraction of the iterations (to ensure progress at every step one would need a larger value of N because of an union bound, but this is unnecessary).

Let us now consider the issue of putting $\mathcal{S}_t$ in near-isotropic position. Let $\hat{\Sigma}_t = \frac{1}{N}\sum_{i=1}^N (X_i - \hat{c}_t)(X_i - \hat{c}_t)^\top$. Rudelson [1999] showed that as long as $N = \widetilde{\Omega}(n)$, one has with high probability (say at least probability $1 - 1/n^2$) that the set $\hat{\Sigma}_t^{-1/2}(\mathcal{S}_t - \hat{c}_t)$ is in near-isotropic position.

Thus it only remains to explain how to sample from a near-isotropic convex set $\mathcal{K}$. This is where random walk ideas come into the picture. The hit-and-run walk[4] is described as follows: at a point $x \in \mathcal{K}$, let $\mathcal{L}$ be a line that goes through x in a direction taken uniformly at random, then move to a point chosen uniformly at random in $\mathcal{L} \cap \mathcal{K}$. Lovász [1998]

---

[4]Other random walks are known for this problem but hit-and-run is the one with the sharpest theoretical guarantees. Curiously we note that one of those walks is closely connected to projected gradient descent, see Bubeck et al. [2015a].

showed that if the starting point of the hit-and-run walk is chosen from a distribution "close enough" to the uniform distribution on $\mathcal{K}$, then after $O(n^3)$ steps the distribution of the last point is ε away (in total variation) from the uniform distribution on $\mathcal{K}$. In the randomized center of gravity method one can obtain a good initial distribution for $\mathcal{S}_t$ by using the distribution that was obtained for $\mathcal{S}_{t-1}$. In order to initialize the entire process correctly we start here with $\mathcal{S}_1 = [-L, L]^n \supset \mathcal{X}$ (in Section 2.1 we used $\mathcal{S}_1 = \mathcal{X}$), and thus we also have to use a *separation oracle* at iterations where $\hat{c}_t \notin \mathcal{X}$, just like we did for the ellipsoid method (see Section 2.2).

Wrapping up the above discussion, we showed (informally) that to attain an ε-optimal point with the randomized center of gravity method one needs: $\widetilde{O}(n)$ iterations, each iterations requires $\widetilde{O}(n)$ random samples from $\mathcal{S}_t$ (in order to put it in isotropic position) as well as a call to either the separation oracle or the first order oracle, and each sample costs $\widetilde{O}(n^3)$ steps of the random walk. Thus overall one needs $\widetilde{O}(n)$ calls to the separation oracle and the first order oracle, as well as $\widetilde{O}(n^5)$ steps of the random walk.

Acknowledgements

References

A. Agarwal and L. Bottou. A lower bound for the optimization of finite sums. *Arxiv preprint arXiv:1410.0723*, 2014.

Z. Allen-Zhu and L. Orecchia. Linear coupling: An ultimate unification of gradient and mirror descent. *Arxiv preprint arXiv:1407.1537*, 2014.

K. M. Anstreicher. Towards a practical volumetric cutting plane method for convex programming. *SIAM Journal on Optimization*, 9(1):190–206, 1998.

J.Y Audibert, S. Bubeck, and R. Munos. Bandit view on noisy optimization. In S. Sra, S. Nowozin, and S. Wright, editors, *Optimization for Machine Learning*. MIT press, 2011.

J.Y. Audibert, S. Bubeck, and G. Lugosi. Regret in online combinatorial optimization. *Mathematics of Operations Research*, 39:31–45, 2014.

F. Bach. Learning with submodular functions: A convex optimization perspective. *Foundations and Trends® in Machine Learning*, 6(2-3):145–373, 2013.

F. Bach and E. Moulines. Non-strongly-convex smooth stochastic approximation with convergence rate o(1/n). In *Advances in Neural Information Processing Systems (NIPS)*, 2013.

F. Bach, R. Jenatton, J. Mairal, and G. Obozinski. Optimization with sparsity-inducing penalties. *Foundations and Trends® in Machine Learning*, 4(1):1–106, 2012.

B. Barak. Sum of squares upper bounds, lower bounds, and open questions. Lecture Notes, 2014.

A. Beck and M. Teboulle. Mirror Descent and nonlinear projected subgradient methods for convex optimization. *Operations Research Letters*, 31(3):167–175, 2003.

A. Beck and M. Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, 2(1):183–202, 2009.

A. Ben-Tal and A. Nemirovski. *Lectures on modern convex optimization: analysis, algorithms, and engineering applications.* Society for Industrial and Applied Mathematics (SIAM), 2001.

D. Bertsimas and S. Vempala. Solving convex programs by random walks. *Journal of the ACM*, 51:540–556, 2004.

S. Boyd and L. Vandenberghe. *Convex Optimization.* Cambridge University Press, 2004.

S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine Learning*, 3(1):1–122, 2011.

S. Bubeck. Introduction to online optimization. Lecture Notes, 2011.

S. Bubeck and N. Cesa-Bianchi. Regret analysis of stochastic and nonstochastic multi-armed bandit problems. *Foundations and Trends® in Machine Learning*, 5(1):1–122, 2012.

S. Bubeck and R. Eldan. The entropic barrier: a simple and optimal universal self-concordant barrier. *Arxiv preprint arXiv:1412.1587*, 2014.

S. Bubeck, R. Eldan, and J. Lehec. Sampling from a log-concave distribution with projected langevin monte carlo. *Arxiv preprint arXiv:1507.02564*, 2015a.

S. Bubeck, Y.-T. Lee, and M. Singh. A geometric alternative to nesterov's accelerated gradient descent. *Arxiv preprint arXiv:1506.08187*, 2015b.

E. Candès and B. Recht. Exact matrix completion via convex optimization. *Foundations of Computational mathematics*, 9(6):717–772, 2009.

A. Cauchy. Méthode générale pour la résolution des systemes d'équations simultanées. *Comp. Rend. Sci. Paris*, 25(1847):536–538, 1847.

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

A. Chambolle and T. Pock. A first-order primal-dual algorithm for convex problems with applications to imaging. *Journal of Mathematical Imaging and Vision*, 40(1):120–145, 2011.

K. Clarkson, E. Hazan, and D. Woodruff. Sublinear optimization for machine learning. *Journal of the ACM*, 2012.

A. Conn, K. Scheinberg, and L. Vicente. *Introduction to Derivative-Free Optimization.* Society for Industrial and Applied Mathematics (SIAM), 2009.

T. M. Cover. 1990 shannon lecture. *IEEE information theory society newsletter*, 42(4), 1992.

A. d'Aspremont. Smooth optimization with approximate gradient. *SIAM Journal on Optimization*, 19(3):1171–1183, 2008.

A. Defazio, F. Bach, and S. Lacoste-Julien. Saga: A fast incremental gradient method with support for non-strongly convex composite objectives. In *Advances in Neural Information Processing Systems (NIPS)*, 2014.

O. Dekel, R. Gilad-Bachrach, O. Shamir, and L. Xiao. Optimal distributed online prediction using mini-batches. *Journal of Machine Learning Research*, 13:165–202, 2012.

J. Duchi, S. Shalev-Shwartz, Y. Singer, and A. Tewari. Composite objective mirror descent. In *Proceedings of the 23rd Annual Conference on Learning Theory (COLT)*, 2010.

J. C. Dunn and S. Harshbarger. Conditional gradient algorithms with open loop step size rules. *Journal of Mathematical Analysis and Applications*, 62 (2):432–444, 1978.

M. Frank and P. Wolfe. An algorithm for quadratic programming. *Naval research logistics quarterly*, 3(1-2):95–110, 1956.

M. P. Friedlander and P. Tseng. Exact regularization of convex programs. *SIAM Journal on Optimization*, 18(4):1326–1350, 2007.

M. Goemans and D. Williamson. Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. *Journal of the ACM*, 42(6):1115–1145, 1995.

M. D. Grigoriadis and L. G. Khachiyan. A sublinear-time randomized approximation algorithm for matrix games. *Operations Research Letters*, 18: 53–58, 1995.

B. Grünbaum. Partitions of mass-distributions and of convex bodies by hyperplanes. *Pacific J. Math*, 10(4):1257–1261, 1960.

T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning.* Springer, 2001.

E. Hazan. The convex optimization approach to regret minimization. In S. Sra, S. Nowozin, and S. Wright, editors, *Optimization for Machine Learning*, pages 287–303. MIT press, 2011.

M. Jaggi. Revisiting frank-wolfe: Projection-free sparse convex optimization. In *Proceedings of the 30th International Conference on Machine Learning (ICML)*, pages 427–435, 2013.

P. Jain, P. Netrapalli, and S. Sanghavi. Low-rank matrix completion using alternating minimization. In *Proceedings of the Forty-fifth Annual ACM Symposium on Theory of Computing*, STOC '13, pages 665–674, 2013.

R. Johnson and T. Zhang. Accelerating stochastic gradient descent using predictive variance reduction. In *Advances in Neural Information Processing Systems (NIPS)*, 2013.

L. K. Jones. A simple lemma on greedy approximation in hilbert space and convergence rates for projection pursuit regression and neural network training. *Annals of Statistics*, pages 608–613, 1992.

A. Juditsky and A. Nemirovski. First-order methods for nonsmooth convex large-scale optimization, i: General purpose methods. In S. Sra, S. Nowozin, and S. Wright, editors, *Optimization for Machine Learning*, pages 121–147. MIT press, 2011a.

A. Juditsky and A. Nemirovski. First-order methods for nonsmooth convex large-scale optimization, ii: Utilizing problem's structure. In S. Sra, S. Nowozin, and S. Wright, editors, *Optimization for Machine Learning*, pages 149–183. MIT press, 2011b.

N. Karmarkar. A new polynomial-time algorithm for linear programming. *Combinatorica*, 4:373–395, 1984.

S. Lacoste-Julien, M. Schmidt, and F. Bach. A simpler approach to obtaining an o (1/t) convergence rate for the projected stochastic subgradient method. *arXiv preprint arXiv:1212.2002*, 2012.

N. Le Roux, M. Schmidt, and F. Bach. A stochastic gradient method with an exponential convergence rate for strongly-convex optimization with finite training sets. In *Advances in Neural Information Processing Systems (NIPS)*, 2012.

Y.-T. Lee and A. Sidford. Path finding i :solving linear programs with ÃŢ(sqrt(rank)) linear system solves. *Arxiv preprint arXiv:1312.6677*, 2013.

Y.-T. Lee, A. Sidford, and S. C.-W Wong. A faster cutting plane method and its implications for combinatorial and convex optimization. *abs/1508.04874*, 2015.

A. Levin. On an algorithm for the minimization of convex functions. In *Soviet Mathematics Doklady*, volume 160, pages 1244–1247, 1965.

L. Lovász. Hit-and-run mixes fast. *Math. Prog.*, 86:443–461, 1998.

G. Lugosi. Comment on: ℓ_1-penalization for mixture regression models. *Test*, 19(2):259–263, 2010.

N. Maculan and G. G. de Paula. A linear-time median-finding algorithm for projecting a vector on the simplex of rn. *Operations research letters*, 8(4): 219–222, 1989.

A. Nemirovski. Orth-method for smooth convex optimization. *Izvestia AN SSSR, Ser. Tekhnicheskaya Kibernetika*, 2, 1982.

A. Nemirovski. Information-based complexity of convex programming. *Lecture Notes*, 1995.

A. Nemirovski. Prox-method with rate of convergence o (1/t) for variational inequalities with lipschitz continuous monotone operators and smooth convex-concave saddle point problems. *SIAM Journal on Optimization*, 15(1):229–251, 2004a.

A. Nemirovski. Interior point polynomial time methods in convex programming. *Lecture Notes*, 2004b.

A. Nemirovski and D. Yudin. *Problem Complexity and Method Efficiency in Optimization*. Wiley Interscience, 1983.

Y. Nesterov. A method of solving a convex programming problem with convergence rate o$(1/k^2)$. *Soviet Mathematics Doklady*, 27(2):372–376, 1983.

Y. Nesterov. Quality of semidefinite relaxation for nonconvex quadratic optimization. CORE Discussion Papers 1997019, Université catholique de Louvain, Center for Operations Research and Econometrics (CORE), 1997.

Y. Nesterov. *Introductory lectures on convex optimization: A basic course*. Kluwer Academic Publishers, 2004a.

Y. Nesterov. Smooth minimization of non-smooth functions. *Mathematical programming*, 103(1):127–152, 2004b.

Y. Nesterov. Gradient methods for minimizing composite objective function. Core discussion papers, Université catholique de Louvain, Center for Operations Research and Econometrics (CORE), 2007.

Y. Nesterov. Efficiency of coordinate descent methods on huge-scale optimization problems. *SIAM Journal on Optimization*, 22:341–362, 2012.

Y. Nesterov and A. Nemirovski. *Interior-point polynomial algorithms in convex programming*. Society for Industrial and Applied Mathematics (SIAM), 1994.

D. Newman. Location of the maximum on unimodal surfaces. *Journal of the ACM*, 12(3):395–398, 1965.

J. Nocedal and S. J. Wright. *Numerical Optimization.* Springer, 2006.

N. Parikh and S. Boyd. Proximal algorithms. *Foundations and Trends® in Optimization*, 1(3):123–231, 2013.

A. Rakhlin. Lecture notes on online learning. 2009.

J. Renegar. *A mathematical view of interior-point methods in convex optimization*, volume 3. Siam, 2001.

P. Richtárik and M. Takác. Parallel coordinate descent methods for big data optimization. *Arxiv preprint arXiv:1212.0873*, 2012.

H. Robbins and S. Monro. A stochastic approximation method. *Annals of Mathematical Statistics*, 22:400–407, 1951.

R. Rockafellar. *Convex Analysis.* Princeton University Press, 1970.

M. Rudelson. Random vectors in the isotropic position. *Journal of Functional Analysis*, 164:60–72, 1999.

M. Schmidt, N. Le Roux, and F. Bach. Convergence rates of inexact proximal-gradient methods for convex optimization. In *Advances in neural information processing systems*, pages 1458–1466, 2011.

B. Schölkopf and A. Smola. *Learning with kernels.* MIT Press, 2002.

S. Shalev-Shwartz and S. Ben-David. *Understanding Machine Learning: From Theory to Algorithms.* Cambridge University Press, 2014.

S. Shalev-Shwartz and T. Zhang. Stochastic dual coordinate ascent methods for regularized loss minimization. *Journal of Machine Learning Research*, 14:567–599, 2013a.

S. Shalev-Shwartz and T. Zhang. Accelerated mini-batch stochastic dual coordinate ascent. In *Advances in Neural Information Processing Systems (NIPS)*, 2013b.

W. Su, S. Boyd, and E. Candès. A differential equation for modeling nesterov's accelerated gradient method: Theory and insights. In *Advances in Neural Information Processing Systems (NIPS)*, 2014.

R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, 58(1):pp. 267–288, 1996.

P. Tseng. On accelerated proximal gradient methods for convex-concave optimization. 2008.

A. Tsybakov. Optimal rates of aggregation. In *Conference on Learning Theory (COLT)*, pages 303–313. 2003.

P. M. Vaidya. A new algorithm for minimizing convex functions over convex sets. In *Foundations of Computer Science, 1989., 30th Annual Symposium on*, pages 338–343, 1989.

P. M. Vaidya. A new algorithm for minimizing convex functions over convex sets. *Mathematical programming*, 73(3):291–341, 1996.

S. J. Wright, R. D. Nowak, and M. A. T. Figueiredo. Sparse reconstruction by separable approximation. *IEEE Transactions on Signal Processing*, 57 (7):2479–2493, 2009.

L. Xiao. Dual averaging methods for regularized stochastic learning and online optimization. *Journal of Machine Learning Research*, 11:2543–2596, 2010.

Y. Zhang and L. Xiao. Stochastic primal-dual coordinate method for regularized empirical risk minimization. *Arxiv preprint arXiv:1409.3257*, 2014.