# Chapter 5

# Optimization for machine learning

**Chapter summary**

- Gradient descent: the workhorse first-order algorithm for optimization, which converges exponentially fast for well-conditioned convex problems.
- Stochastic gradient descent (SGD): the workhorse first-order algorithm for large-scale machine learning, which converges as $1/t$ or $1/\sqrt{t}$, where $t$ is the number of iterations.
- Generalization bounds through stochastic gradient descent: with only a single pass on the data, there is no risk of overfitting, and we obtain generalization bounds for unseen data.
- Variance reduction: when minimizing strongly-convex finite sums, this class of algorithms is exponentially convergent while having a small iteration complexity.

In this chapter, we present optimization algorithms based on gradient descent and analyze their performance, mainly on convex objective functions. We will consider generic algorithms that have applications beyond machine learning and algorithms dedicated to machine learning (such as stochastic gradient methods). See Nesterov (2018); Bubeck (2015) for further details.

## 5.1 Optimization in machine learning

In supervised machine learning, we are given $n$ i.i.d. samples $(x_i, y_i)$, $i = 1, \dots, n$ of a couple of random variables $(x, y)$ on $\mathcal{X} \times \mathcal{Y}$ and the goal is to find a predictor $f : \mathcal{X} \to \mathbb{R}$

with a small risk on unseen data

$$\mathcal{R}(f) := \mathbb{E}[\ell(y, f(x))],$$

where $\ell : \mathcal{Y} \times \mathbb{R} \to \mathbb{R}$ is a loss function. This loss is typically convex in the second argument (e.g., square loss or logistic loss, see Chapter 4), which is often considered a weak assumption.

In the empirical risk minimization approach described in Chapter 4, we choose the predictor by minimizing the empirical risk over a parameterized set of predictors, potentially with regularization. For a parameterization $\{f_\theta\}_{\theta \in \mathbb{R}^d}$ and a regularizer $\Omega : \mathbb{R}^d \to \mathbb{R}$ (e.g., $\Omega(\theta) = \|\theta\|_2^2$ or $\Omega(\theta) = \|\theta\|_1$), this requires to minimize the function

$$F(\theta) := \frac{1}{n} \sum_{i=1}^{n} \ell(y_i, f_\theta(x_i)) + \Omega(\theta). \tag{5.1}$$

In optimization, the function $F : \mathbb{R}^d \to \mathbb{R}$ is called the *objective function*.

In general, the minimizer has no closed form. Even when it has one (e.g., linear predictor and square loss in Chapter 3), it could be expensive to compute for large problems. We thus resort to iterative algorithms.

**Accuracy of iterative algorithms.** Solving optimization problems with high accuracy is computationally expensive, and the goal is not to minimize the training objective but the error on unseen data.

Then, which accuracy is satisfying in machine learning? If the algorithm returns $\hat{\theta}$, and we define $\theta_* \in \arg\min_\theta \mathcal{R}(f_\theta)$, we have the risk decomposition from Section 2.3.2 (where the approximation error due to the use of a specific set of models $f_\theta$, $\theta \in \Theta$ is ignored):

$$\mathcal{R}(f_{\hat{\theta}}) - \inf_{\theta \in \mathbb{R}^d} \mathcal{R}(f_\theta) = \underbrace{\big\{\mathcal{R}(f_{\hat{\theta}}) - \widehat{\mathcal{R}}(f_{\hat{\theta}})\big\}}_{\text{estimation error}} + \underbrace{\big\{\widehat{\mathcal{R}}(f_{\hat{\theta}}) - \widehat{\mathcal{R}}(f_{\theta_*})\big\}}_{\text{optimization error}} + \underbrace{\big\{\widehat{\mathcal{R}}(f_{\theta_*}) - \mathcal{R}(f_{\theta_*})\big\}}_{\text{estimation error}},$$

where we added the second term, the *optimization error*, which will always be negative if $\hat{\theta}$ is the minimizer of $\widehat{\mathcal{R}}$, and is usually upper-bounded by $\widehat{\mathcal{R}}(f_{\hat{\theta}}) - \inf_{\theta \in \Theta} \widehat{\mathcal{R}}(f_\theta)$. It is thus sufficient to reach an optimization accuracy of the order of the estimation error (usually of the order $O(1/\sqrt{n})$ or $O(1/n)$, see Chapter 3 and Chapter 4). Note that for machine learning, the optimization error defined above corresponds to characterizing approximate solutions through function values. While this will be one central focal point in this chapter, we will consider other performance measures.

In this chapter, we will first look at minimization without focusing on machine learning problems (Section 5.2), with both smooth and non-smooth objective functions. We will then look at stochastic gradient descent in Section 5.4, which can be used to obtain bounds on both the training and testing risks. We then briefly present adaptive methods in

Section 5.4.2, bias-variance decompositions for least-squares in Section 5.4.3, and variance reduction in Section 5.4.4.

⚠ The notation $\theta_*$ may mean different things in optimization and machine learning: minimizer of the *regularized empirical risk*, or minimizer of the *expected risk*. For the sake of clarity, we will use the notation $\eta_*$ for the minimizer of empirical (potentially regularized) risk, that is, when we look at optimization problems, and $\theta_*$ for the minimizer of the expected risk, that is, when we look at statistical problems.

⚠ Sometimes, we mention solving a problem with *high* precision. This corresponds to a *low* optimization error.

In this chapter, we primarily focus on gradient descent methods for convex optimization problems, which, in learning terms, correspond to predictors that are linear in their parameter (an assumption that will be relaxed in subsequent chapters) and a convex loss function such as the logistic loss or the square loss. We first consider so-called "batch methods" that do not use the finite sum structure of the objective function in Eq. (5.1) before moving on to the stochastic gradient method that does take into account this structure for enhanced computational efficiency.

## 5.2 Gradient descent

Suppose we want to solve, for a function $F : \mathbb{R}^d \to \mathbb{R}$, the optimization problem

$$\min_{\theta \in \mathbb{R}^d} F(\theta).$$

We assume that we are given access to certain "oracles": the *$k$-th-order oracle* corresponds to the access to: $\theta \mapsto (F(\theta), F'(\theta), \dots, F^{(k)}(\theta))$, that is all partial derivatives up to order $k$. All algorithms will call these oracles; thus, their computational complexity will depend directly on the complexity of this oracle. For example, for least-squares with a design matrix in $\mathbb{R}^{n \times d}$, computing a single gradient of the empirical risk costs $O(nd)$.

In this section, for the algorithms and proofs, we do not assume that the function $F$ is the regularized empirical risk, but this situation will be our motivating example throughout. We will study the following first-order algorithm.

**Algorithm 5.1 (Gradient descent (GD))** *Pick $\theta_0 \in \mathbb{R}^d$ and for $t \geqslant 1$, let*

$$\theta_t = \theta_{t-1} - \gamma_t F'(\theta_{t-1}), \tag{5.2}$$

*for a well (potentially adaptively) chosen step-size sequence $(\gamma_t)_{t \geqslant 1}$.*

For machine learning problems where the empirical risk is minimized, computing the gradient $F'(\theta_{t-1})$ requires computing all gradients of $\theta \mapsto \ell(y_i, f_\theta(x_i))$, and averaging them.

There are many ways to choose the step-size $\gamma_t$, either constant, decaying, or through a line search.[1] In practice, using some form of line search is usually advantageous and is implemented in most applications. See Armijo (1966) and Goldstein (1962) for convergence guarantees with typical procedures. In this chapter, since we want to focus on the simplest algorithms and proofs, we will focus on step-sizes that depend explicitly on problem constants and sometimes on the iteration number. When gradients are not available, gradient estimates may be built from function values (see, e.g., Nesterov and Spokoiny, 2017, and Chapter 11). Note that the differences between convergence rates with and without line searches are generally not significant (see Exercise 5.2 below for quadratic functions). At the same time, practical behavior is significantly improved with line search.

We start with the simplest example, namely convex quadratic functions, where the most important concepts already appear.

### 5.2.1 Simplest analysis: ordinary least-squares

We start with a case where the analysis is explicit: ordinary least squares (see Chapter 3 for the statistical analysis). Let $\Phi \in \mathbb{R}^{n\times d}$ be the design matrix and $y \in \mathbb{R}^n$ the vector of responses. Least-squares estimation amounts to finding a minimizer $\eta_*$ of

$$F(\theta) = \frac{1}{2n}\|\Phi\theta - y\|_2^2. \tag{5.3}$$

⚠ A factor of $\frac{1}{2}$ has been added compared to Chapter 3 to get nicer looking gradients.

The gradient of $F$ is $F'(\theta) = \frac{1}{n}\Phi^\top(\Phi\theta - y) = \frac{1}{n}\Phi^\top\Phi\theta - \frac{1}{n}\Phi^\top y$. Thus, denoting $H = \frac{1}{n}\Phi^\top\Phi \in \mathbb{R}^{d\times d}$ the Hessian matrix (equal for all $\theta$, denoted $\widehat{\Sigma}$ in Chapter 3), minimizers $\eta_*$ are characterized by

$$H\eta_* = \frac{1}{n}\Phi^\top y.$$

Since $\frac{1}{n}\Phi^\top y \in \mathbb{R}^d$ is in the column space of $H$, there is always a minimizer, but unless $H$ is invertible, the minimizer is not unique. But all minimizers $\eta_*$ have the same function value $F(\eta_*)$, and we have, from a simple exact Taylor expansion (and using $F'(\eta_*) = 0$):

$$F(\theta) - F(\eta_*) = F'(\eta_*)^\top(\theta - \eta_*) + \frac{1}{2}(\theta - \eta_*)^\top H(\theta - \eta_*) = \frac{1}{2}(\theta - \eta_*)^\top H(\theta - \eta_*).$$

Two quantities will be important in the following developments: the largest eigenvalue $L$ and the smallest eigenvalue $\mu$ of the Hessian matrix $H$. As a consequence of the convexity of the objective, we have $0 \leqslant \mu \leqslant L$. We denote by $\kappa = \frac{L}{\mu} \geqslant 1$ the *condition number*.

Note that for least-squares, $\mu$ is the lowest eigenvalue of the non-centered empirical covariance matrix and that it is zero as soon as $d > n$, and, in most practical cases, *very* small. When adding a regularizer $\frac{\lambda}{2}\|\theta\|_2^2$ (like in ridge regression), then $\mu \geqslant \lambda$ (but then $\lambda$ typically decreases with $n$, often between $\frac{1}{\sqrt{n}}$ and $\frac{1}{n}$, see Chapter 7 for more details).

[1] See, e.g., https://en.wikipedia.org/wiki/Line_search.

**Closed-form expression.** Gradient descent iterates with fixed step-size $\gamma_t = \gamma$ can be computed in closed form:

$$\theta_t = \theta_{t-1} - \gamma F'(\theta_{t-1}) = \theta_{t-1} - \gamma\big[\frac{1}{n}\Phi^\top(\Phi\theta_{t-1} - y)\big] = \theta_{t-1} - \gamma H(\theta_{t-1} - \eta_*),$$

leading to

$$\theta_t - \eta_* = \theta_{t-1} - \eta_* - \gamma H(\theta_{t-1} - \eta_*) = (I - \gamma H)(\theta_{t-1} - \eta_*),$$

that is, we have a linear recursion, and we can unroll the recursion and now write

$$\theta_t - \eta_* = (I - \gamma H)^t(\theta_0 - \eta_*).$$

We can now look at various measures of performance:

$$\begin{aligned} \|\theta_t - \eta_*\|_2^2 &= (\theta_0 - \eta_*)^\top (I - \gamma H)^{2t}(\theta_0 - \eta_*) \\ F(\theta_t) - F(\eta_*) &= \frac{1}{2}(\theta_0 - \eta_*)^\top (I - \gamma H)^{2t} H (\theta_0 - \eta_*). \end{aligned}$$

The two optimization performance measures differ by the presence of the Hessian matrix $H$ in the measure based on function values.

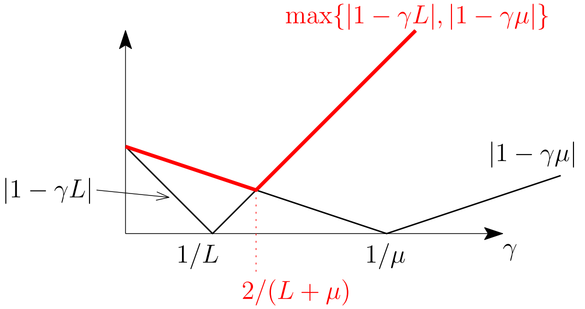
**Convergence in distance to the minimizer.** If we hope to have $\|\theta_t - \eta_*\|_2^2$ going to zero, we need to have a single minimizer $\eta_*$, and thus $H$ has to be invertible, that is $\mu > 0$. Given the form of $\|\theta_t - \eta_*\|_2^2$, we simply need to bound the eigenvalues of $(I - \gamma H)^{2t}$ (since for a positive semi-definite matrix $M$, $u^\top M u \leqslant \lambda_{\max}(M)\|u\|_2^2$ for all vectors $u$).

The eigenvalues of $(I - \gamma H)^{2t}$ are exactly $(1 - \gamma\lambda)^{2t}$ for $\lambda$ an eigenvalue of $H$ (all of them are in the interval $[\mu, L]$). Thus all the eigenvalues of $(I - \gamma H)^{2t}$ have magnitude less than

$$\Big(\max_{\lambda \in [\mu, L]} |1 - \gamma\lambda|\Big)^{2t}.$$

We can then have several strategies for choosing the step-size $\gamma$:

- Optimal choice: one can check that minimizing $\max_{\lambda \in [\mu, L]} |1 - \gamma\lambda|$ is done by setting $\gamma = 2/(\mu + L)$, with an optimal value equal to $\frac{\kappa - 1}{\kappa + 1} = 1 - \frac{2}{\kappa + 1} \in (0, 1)$. See geometric "proof" below.



- Choice independent of $\mu$: with the simpler (slightly smaller) choice $\gamma = 1/L$, we get $\max_{\lambda \in [\mu, L]} |1 - \gamma\lambda| = (1 - \frac{\mu}{L}) = (1 - \frac{1}{\kappa})$, which is only slightly larger than the value for the optimal choice. Note that all step-sizes strictly less than $2/L$ will lead to exponential convergence.

For example, with the weaker choice $\gamma = 1/L$, we get:

$$\|\theta_t - \eta_*\|_2^2 \leqslant \Big(1 - \frac{1}{\kappa}\Big)^{2t} \|\theta_0 - \eta_*\|_2^2,$$

which is often referred to as exponential, geometric, or linear convergence.

⚠ The denomination "linear" is sometimes confusing and corresponds to a number of significant digits that grows linearly with the number of iterations.

We can further bound $\big(1 - \frac{1}{\kappa}\big)^{2t} \leqslant \exp(-1/\kappa)^{2t} = \exp(-2t/\kappa)$, and thus the characteristic time of convergence is of order $\kappa$. We will often make the calculation $\varepsilon = \exp(-2t/\kappa) \Leftrightarrow t = \frac{\kappa}{2} \log \frac{1}{\varepsilon}$. Thus, for a relative reduction of squared distance to the optimum of $\varepsilon$, we need at most $t = \frac{\kappa}{2} \log \frac{1}{\varepsilon}$ iterations.

For $\kappa = +\infty$, the result remains true but simply says that for all minimizers $\|\theta_t - \eta_*\|_2^2 \leqslant \|\theta_0 - \eta_*\|_2^2$, which is a good sign (the algorithm does not move away from minimizers) but not indicative of any form of convergence. We will need to use a different criterion.

**Convergence in function values.** Using the same step-size $\gamma = 1/L$ as above, and using the upper-bound on eigenvalues of $(I - \gamma H)^{2t}$ (which are all less than $(1 - 1/\kappa)^{2t}$), we get

$$F(\theta_t) - F(\eta_*) \leqslant \Big(1 - \frac{1}{\kappa}\Big)^{2t} [F(\theta_0) - F(\eta_*)] \leqslant \exp(-2t/\kappa)[F(\theta_0) - F(\eta_*)]. \tag{5.4}$$

When $\kappa < \infty$ (that is, $\mu > 0$), we also obtain linear convergence for this criterion, but when $\kappa = \infty$, this is non-informative.

To obtain a convergence rate, we will need to bound the eigenvalues of $(I - \gamma H)^{2t} H$ instead of $(I - \gamma H)^{2t}$. The key difference is that for eigenvalues $\lambda$ of $H$ which are close to zero, $(1 - \gamma\lambda)^{2t}$ does not have a strong contracting effect, but they count less as they are multiplied by $\lambda$ in the bound.

We can make this trade-off precise, for $\gamma \leqslant 1/L$, as

$$\begin{aligned} \big|\lambda(1 - \gamma\lambda)^{2t}\big| &\leqslant \lambda \exp(-\gamma\lambda)^{2t} = \lambda \exp(-2t\gamma\lambda) \\ &= \frac{1}{2t\gamma} 2t\gamma\lambda \exp(-2t\gamma\lambda) \leqslant \frac{1}{2t\gamma} \sup_{\alpha \geqslant 0} \alpha \exp(-\alpha) = \frac{1}{2et\gamma} \leqslant \frac{1}{4t\gamma}, \end{aligned}$$

where we used that $\alpha e^{-\alpha}$ is maximized over $\mathbb{R}_+$ at $\alpha = 1$ (as the derivative is $e^{-\alpha}(1 - \alpha)$).

This leads to, with the largest step-size $\gamma = 1/L$:

$$F(\theta_t) - F(\eta_*) \leqslant \frac{1}{8t\gamma} \|\theta_0 - \eta_*\|_2^2 = \frac{L}{8t} \|\theta_0 - \eta_*\|_2^2. \tag{5.5}$$

We can make the following observations:

- ⚠ The convergence results in $\exp(-2t/\kappa)$ in Eq. (5.4) for invertible Hessians or $1/t$ in general in Eq. (5.5) are only upper-bounds! It is good to understand the gap between the bounds and the actual performance, as this is possible for quadratic objective functions.

  For the exponentially convergent case, the lowest eigenvalue $\mu$ dictates the rate for all eigenvalues. So, if the eigenvalues are well-spread (or if only one eigenvalue is very small), there can be quite a strong discrepancy between the bound and the actual behavior.

  For the rate in $1/t$, the bound in eigenvalues is tight when $t\gamma\lambda$ is of order 1, namely when $\lambda$ is of order $1/(t\gamma)$. Thus, to see an $O(1/t)$ convergence rate in practice, we need to have sufficiently many small eigenvalues. As $t$ grows, we often go to a local linear convergence phase where the smallest non-zero eigenvalue of $H$ kicks in. See the simulations and the exercise below.

  **Exercise 5.1** *Let $\mu_+$ be the smallest non-zero eigenvalue of $H$. Show that gradient descent is linearly convergent with the contracting rate $(1 - \mu_+/L)$.*

- From errors to numbers of iterations: as already mentioned, the bound in Eq. (5.4) says that after $t$ steps, the reduction in suboptimality in function values is multiplied by $\varepsilon = \exp(-2t/\kappa)$. This can be reinterpretated as a need of $t = \frac{\kappa}{2}\log\frac{1}{\varepsilon}$ iterations to reach a relative error $\varepsilon$.

- Can an algorithm having the same access to oracles of $F$ do better?

  If we have access to matrix-vector products with the matrix $\Phi$, then the conjugate gradient algorithm can be used with convergence rates in $\exp(-t/\sqrt{\kappa})$ and $1/t^2$ (see Golub and Loan, 1996). With only access to gradients of $F$ (which is a bit weaker), Nesterov acceleration (see Section 5.2.5 below) will also lead to the same convergence rates, which are then optimal (for a sense to be defined later in this chapter and in more details in Chapter 15).

- Can we extend beyond least-squares? The convergence results above will generalize to convex functions (see Section 5.2.2) but with less direct proofs. Non-convex objectives are discussed in Section 5.2.6.

**Experiments.** We consider two quadratic optimization problems in dimension $d = 1000$, with two different decays of eigenvalues $(\lambda_k)_{k\in\{1,\dots,d\}}$ for the Hessian matrix $H$, one as $1/k$ (in blue below) and one in $1/k^2$ (in red below), and for which we plot in Figure 5.1 the performance for function values, both in semi-logarithm plots (left) and full-logarithm plots (right). For slow decays (blue), we see the linear convergence kicking in (line in the left "semi-log" plot), while for fast decays (red), we obtain a polynomial rate that is not exponential (line in the right "log-log" plot). Note that the bound in Eq. (5.5) is very pessimistic and does not lead to the correct power of $t$ (which, as can be checked as an exercise, should be $1/\sqrt{t}$ for $t$ small enough compared to $d$).

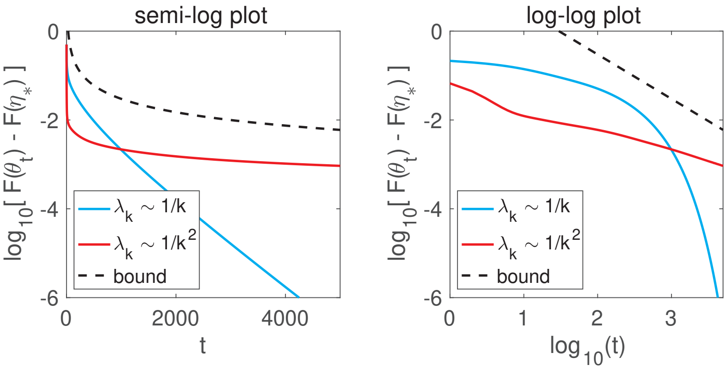


Figure 5.1: Gradient descent on two least-squares problems with step-size $\gamma = 1/L$, and two different sets of eigenvalues $(\lambda_k)_{k\in\{1,\dots,d\}}$ of the Hessian, together with the bound from Eq. (5.5). Left: semi-logarithmic scale. Right: joint logarithmic scale.

**Exercise 5.2** *(exact line search ♦) For the quadratic objective in Eq. (5.3), show that the optimal step-size $\gamma_t$ in Eq. (5.2) is equal to $\gamma_t = \frac{\|F'(\theta_{t-1})\|_2^2}{F'(\theta_{t-1})^\top H F'(\theta_{t-1})}$. Show that when $F$ is strongly-convex, $F(\theta_t) - F(\eta_*) \leqslant \big(\frac{\kappa-1}{\kappa+1}\big)^2\big[F(\theta_{t-1}) - F(\eta_*)\big]$, and compare the rate with constant step-size gradient descent.*
*Hint: prove and use the Kantorovich inequality $\sup_{\|z\|_2=1} z^\top H z z^\top H^{-1} z = \frac{(L+\mu)^2}{4\mu L}$.*

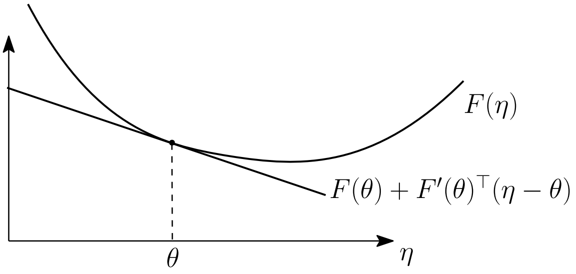
### 5.2.2 Convex functions and their properties

We now wish to analyze GD (and later its stochastic version SGD) in a broader setting. We will always assume convexity, although these algorithms are also used (and can sometimes also be analyzed) when this assumption does not hold (see Section 5.2.6). In other words, convexity is most often used for analysis rather than to define the algorithm.

**Definition 5.1 (Convex function)** *A differentiable function $F : \mathbb{R}^d \to \mathbb{R}$ is said* convex *if and only if*

$$F(\eta) \geqslant F(\theta) + F'(\theta)^\top(\eta - \theta), \qquad \forall \eta, \theta \in \mathbb{R}^d. \tag{5.6}$$

This corresponds to the function $F$ being above its tangent at $\theta$, as illustrated below.



If $f$ is twice-differentiable, this is equivalent to requiring $F''(x) \succcurlyeq 0$, $\forall x \in \mathbb{R}^d$; here $\succcurlyeq$ denotes the semidefinite partial ordering—also called the Löwner order—characterized by $A \succcurlyeq B \Leftrightarrow A - B$ is positive semidefinite, see Boyd and Vandenberghe (2004); Bhatia (2009).

An important consequence that we will use a lot in this chapter is, for all $\theta \in \mathbb{R}^d$ (and using $\eta = \eta_*$)

$$F(\eta_*) \geqslant F(\theta) + F'(\theta)^\top(\eta_* - \theta) \quad \Leftrightarrow \quad F(\theta) - F(\eta_*) \leqslant F'(\theta)^\top(\theta - \eta_*), \tag{5.7}$$

that is, the distance to optimum in function values is upper bounded by a function of the gradient (note that it provides proof that $F'(\theta) = 0$ implies that $\theta$ is a global minimizer of $F$).

A more general definition of convexity (without gradients) is that $\forall \theta, \eta \in \mathbb{R}^d$ and $\alpha \in [0, 1]$,

$$F(\alpha\eta + (1-\alpha)\theta) \leqslant \alpha F(\eta) + (1-\alpha)F(\theta),$$

which generalizes to the usual Jensen's inequality below.[2]

**Proposition 5.1 (Jensen's inequality)** *If $F : \mathbb{R}^d \to \mathbb{R}$ is convex and $\mu$ is a probability measure on $\mathbb{R}^d$, then*

$$F\Big(\int_{\mathbb{R}^d} \theta d\mu(\theta)\Big) \leqslant \int_{\mathbb{R}^d} F(\theta) d\mu(\theta). \tag{5.8}$$

*In words: "the image of the average is smaller than the average of the images".*

⚠ When using Jensen's inequality, be extra careful in the direction of the inequality.

**Exercise 5.3** *Assume that the function $F : \mathbb{R}^d \to \mathbb{R}$ is* strictly convex, *that is, $\forall \theta, \eta \in \mathbb{R}^d$ such that $\theta \neq \eta$, and $\alpha \in (0, 1)$, $F(\alpha\eta + (1-\alpha)\theta) < \alpha F(\eta) + (1-\alpha)F(\theta)$. Show that there is equality in Jensen's inequality in Eq. (5.8) if and only if the random variable $\theta$ is almost surely constant.*

The class of convex functions satisfies the following stability properties (proofs left as an exercise); for more properties on convex functions, see Boyd and Vandenberghe (2004):

- If $(F_j)_{j \in \{1,\dots,m\}}$ are convex and $(\alpha_j)_{j \in \{1,\dots,m\}}$ are nonnegative, then $\sum_{j=1}^m \alpha_j F_j$ and $\max_{j \in \{1,\dots,m\}} F_j$ are convex.
- If $F : \mathbb{R}^d \to \mathbb{R}$ is convex and $A : \mathbb{R}^{d'} \to \mathbb{R}^d$ is linear then $F \circ A : \mathbb{R}^{d'} \to \mathbb{R}$ is convex.
- If $F : \mathbb{R}^{d_1 + d_2} \to \mathbb{R}$ is convex, so is the function $x_1 \mapsto \inf_{x_2 \in \mathbb{R}^{d_2}} F(x_1, x_2)$ on $\mathbb{R}^{d_1}$.

**Classical machine learning example.** Problems of the form in Eq. (5.1) are convex if the loss $\ell$ is convex in the second variable, $f_\theta(x)$ is linear in $\theta$, and $\Omega$ is convex.

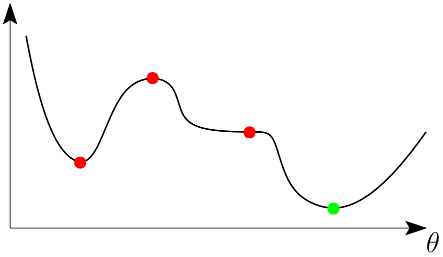
[2] See several applications in `https://francisbach.com/jensen-inequality/`.

**Global optimality from local information.** It is also worth emphasizing the following property (immediate from the definition).

**Proposition 5.2** *Assume that $F : \mathbb{R}^d \to \mathbb{R}$ is convex and differentiable. Then $\eta_* \in \mathbb{R}^d$ is a global minimizer of $F$ if and only if*

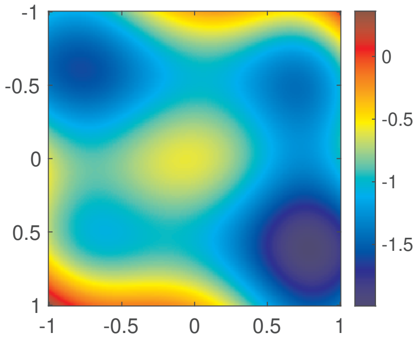
$$F'(\eta_*) = 0.$$

This implies that for convex functions, we only need to look for stationary points. This is *not* the case for potentially non-convex functions. For example, in one dimension below, all red points are stationary points that are not the global minimum (which is in green).



The situation is even more complex in higher dimensions. Note that without convexity assumptions, optimization of Lipschitz-continuous functions will need exponential time in dimension in the worst case (see Section 15.2.2).

**Exercise 5.4** *Identify all stationary points in the function in $\mathbb{R}^2$ depicted below.*



### 5.2.3 Analysis of GD for strongly convex and smooth functions

The analysis of optimization algorithms requires assumptions on the objective functions, like the ones introduced in this section. From these assumptions, additional properties are derived (typically inequalities), and then most convergence proofs look for a “Lyapunov function” (sometimes called a “potential function”) that goes down along the iterations. More precisely, if $V : \mathbb{R}^d \to \mathbb{R}_+$ is such that $V(\theta_t) \leqslant (1-\alpha)V(\theta_{t-1})$, then $V(\theta_t) \leqslant (1-\alpha)^t V(\theta_0)$ and we obtain linear convergence. The art is then to find the appropriate

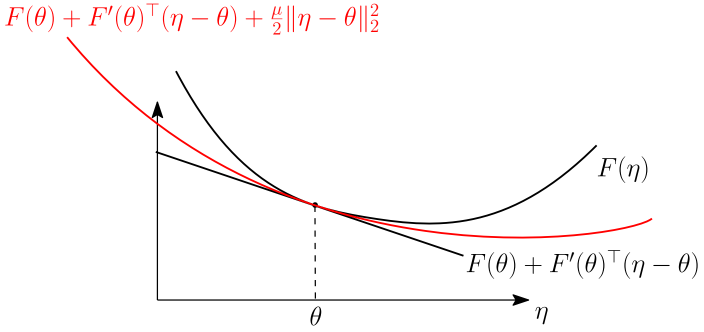
Lyapunov function; for slower convergence rates, weaker forms of decrease for Lyapunov functions will be considered.

We first consider an assumption allowing exponential convergence rates.

**Definition 5.2 (Strong convexity)** *A differentiable function $F$ is said $\mu$-strongly convex, with $\mu > 0$, if and only if*

$$F(\eta) \geqslant F(\theta) + F'(\theta)^\top(\eta - \theta) + \frac{\mu}{2}\|\eta - \theta\|_2^2, \quad \forall \eta, \theta \in \mathbb{R}^d. \tag{5.9}$$

The function $F$ is strongly-convex if and only if the function $F$ is strictly above its tangent and the difference is at least quadratic in the distance to the point where the two coincide. This notably allows us to define quadratic lower bounds on $F$. See below.


$F(\theta) + F'(\theta)^\top(\eta - \theta) + \frac{\mu}{2}\|\eta - \theta\|_2^2$
$F(\eta)$
$F(\theta) + F'(\theta)^\top(\eta - \theta)$
$\theta$
$\eta$

For twice differentiable functions, this is equivalent to $F''(\theta) \succcurlyeq \mu I$ for all $\theta$, that is, all eigenvalues of $F''(\theta)$ are greater than or equal to $\mu$ (see Nesterov, 2018), but non-smooth functions can be strongly-convex, since, as a consequence of the exercise below, we can add $\frac{\mu}{2}\|\cdot\|_2^2$ to any (potentially non-smooth) convex function to make it $\mu$-strongly-convex.

**Exercise 5.5** *Show that $F : \mathbb{R}^d \to \mathbb{R}$ is $\mu$-strongly-convex if and only if the function $\theta \mapsto F(\theta) - \frac{\mu}{2}\|\theta\|_2^2$ is convex.*

**Exercise 5.6** *Show that if $F : \mathbb{R}^d \to \mathbb{R}$ is $\mu$-strongly-convex, then it has a unique minimizer.*

**Exercise 5.7** *Show that the differentiable function $F : \mathbb{R}^d \to \mathbb{R}$ is $\mu$-strongly convex if and only if for all $\theta, \eta \in \mathbb{R}^d$, $\|F'(\theta) - F'(\eta)\|_2 \geqslant \mu\|\theta - \eta\|_2$.*

**Strong convexity through regularization.** When an objective function $F$ is convex, then $F + \frac{\mu}{2}\|\cdot\|_2^2$ is $\mu$-strongly convex (proof left as an exercise). In practice, in machine learning problems with linear models, so that the empirical risk is convex, strong convexity most often comes from the regularizer (and thus $\mu$ decays with $n$), leading to condition numbers that grow with $n$ (typically in $\sqrt{n}$ or $n$). While the regularizer was added in Section 3.6 to improve generalization, we see in this section that it also leads to faster optimization algorithms, showing that statistical and optimization performances are often aligned.

**Łojasiewicz inequality.** Strong convexity implies that $F$ admits a unique minimizer $\eta_*$, which is characterized by $F'(\eta_*) = 0$. Moreover, this guarantees that the gradient is large when a point is far from optimal (in function values):

**Lemma 5.1 (Łojasiewicz inequality)** *If $F$ is differentiable and $\mu$-strongly convex with unique minimizer $\eta_*$, then we have:*

$$\|F'(\theta)\|_2^2 \geqslant 2\mu(F(\theta) - F(\eta_*)), \quad \forall \theta \in \mathbb{R}^d.$$

**Proof** The right-hand side in Definition 5.2 is strongly convex in $\eta$ and minimized with $\tilde{\eta} = \theta - \frac{1}{\mu}F'(\theta)$. Plugging this value into the bound and taking $\eta = \eta_*$ in the left-hand side, we get $F(\eta_*) \geqslant F(\theta) - \frac{1}{\mu}\|F'(\theta)\|_2^2 + \frac{1}{2\mu}\|F'(\theta)\|_2^2 = F(\theta) - \frac{1}{2\mu}\|F'(\theta)\|_2^2$. The conclusion follows by rearranging. ■

Note that while strong convexity is a sufficient condition for the Łojasiewicz inequality, it is unnecessary (see, e.g., Section 12.1.1).

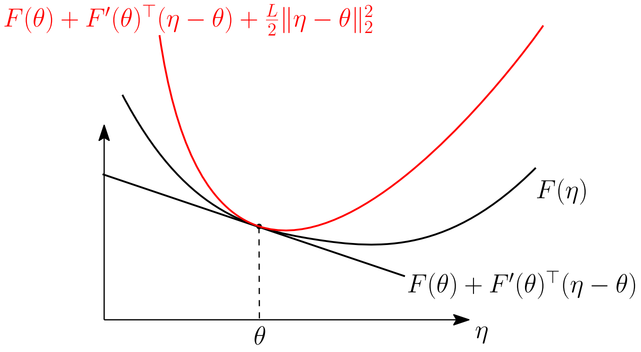
To obtain exponential convergence rates, strong-convexity is typically associated with smoothness, which we now define.

**Definition 5.3 (Smoothness)** *A differentiable function $F$ is said $L$-smooth if and only if*

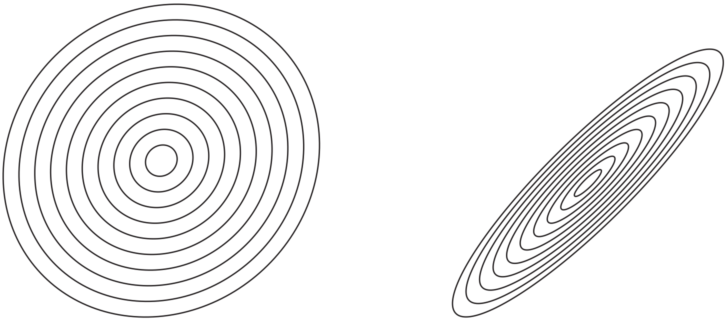
$$|F(\eta) - F(\theta) - F'(\theta)^\top(\eta - \theta)| \leqslant \frac{L}{2}\|\theta - \eta\|^2, \quad \forall \theta, \eta \in \mathbb{R}^d. \tag{5.10}$$

This is equivalent to $F$ having a $L$-Lipschitz-continuous gradient with respect to the $\ell_2$-norm, i.e., $\|F'(\theta) - F'(\eta)\|_2^2 \leqslant L^2\|\theta - \eta\|_2^2$, $\forall \theta, \eta \in \mathbb{R}^d$. For twice differentiable functions, this is equivalent to $-LI \preccurlyeq F''(\theta) \preccurlyeq LI$ (see Nesterov, 2018). If the function is also $\mu$-strongly convex, then all eigenvalues of all Hessians are in the interval $[\mu, L]$.

Note that when $F$ is convex and $L$-smooth, we have a quadratic upper bound that is tight at any given point (strong convexity implies the corresponding lower bound with $L$ replaced by $\mu$). See below.


$F(\theta) + F'(\theta)^\top(\eta - \theta) + \frac{L}{2}\|\eta - \theta\|_2^2$
$F(\eta)$
$F(\theta) + F'(\theta)^\top(\eta - \theta)$
$\theta$
$\eta$

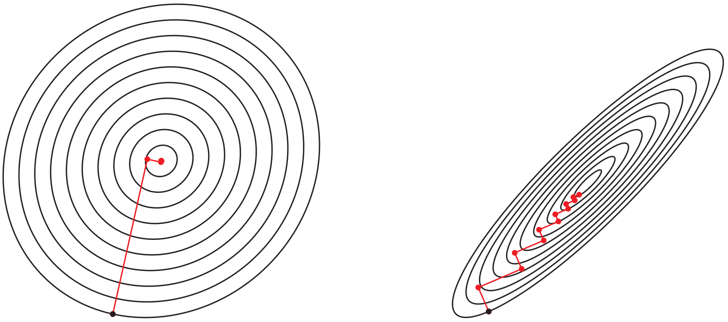
When a function is both smooth and strongly convex, we denote by $\kappa = L/\mu \geqslant 1$ its condition number (we recover the definition of Section 5.2.1 for quadratic functions). See examples below of level sets of functions with varying condition numbers: the condition number impacts the shapes of the level sets.



(small $\kappa = L/\mu$) (large $\kappa = L/\mu$)

The performance of gradient descent will depend on this condition number (see steepest descent below, that is, gradient descent with exact line search): with a small condition number (left), we get fast convergence, while for a large condition number (right), we get oscillations.

**Exercise 5.8** *(♦) We consider the angle $\alpha$ between the descent direction $-F'(\theta)$ and the deviation to optimum $\theta - \eta_*$, defined through $\cos\alpha = \frac{F'(\theta)^\top(\theta-\eta_*)}{\|F'(\theta)\|\cdot\|\theta-\eta_*\|_2}$. Show that for a $\mu$-strongly-convex, $L$-smooth quadratic function, $\cos\alpha \geqslant \frac{2\sqrt{\mu L}}{L+\mu}$ (hint: prove and use the Kantorovich inequality $\sup_{\|z\|_2=1} z^\top H z z^\top H^{-1} z = \frac{(L+\mu)^2}{4\mu L}$). (♦♦) Show that the same result holds without the assumption that $F$ is quadratic (hint: use the co-coercivity of the function $\theta \mapsto F(\theta) - \frac{\mu}{2}\|\theta\|_2^2$, from Prop. 5.4).*



(small $\kappa = L/\mu$) (large $\kappa = L/\mu$)

For machine learning problems, such as linear predictions and smooth losses (square or logistic), we have smooth problems. If we use a squared $\ell_2$-regularizer $\frac{\mu}{2}\|\cdot\|_2^2$ , we get a $\mu$-strongly convex problem. Note that when using regularization, as explained in Chapters 3 and 4, the value of $\mu$ decays with $n$, typically between $1/n$ and $1/\sqrt{n}$, leading to condition numbers between $\sqrt{n}$ and $n$.

In this context, gradient descent on the empirical risk is often called a "batch" technique because all the data points are accessed at every iteration.

In the next proposition, we show that gradient descent converges exponentially for such smooth and strongly-convex problems, thus extending the result for quadratic functions from Section 5.2.1.

**Proposition 5.3 (Convergence of GD for smooth strongly-convex functions)** *Assume that $F$ is $L$-smooth and $\mu$-strongly convex. Choosing $\gamma_t = 1/L$, the iterates $(\theta_t)_{t \geqslant 0}$ of GD on $F$ satisfy*

$$F(\theta_t) - F(\eta_*) \leqslant \Big(1 - \frac{1}{\kappa}\Big)^t (F(\theta_0) - F(\eta_*)) \leqslant \exp(-t/\kappa)(F(\theta_0) - F(\eta_*)).$$

**Proof** By the smoothness inequality in Eq. (5.10) applied to $\theta_{t-1}$ and $\theta_{t-1} - F'(\theta_{t-1})/L$, we have the following descent property, with $\gamma_t = 1/L$,

$$\begin{aligned} F(\theta_t) &= F\big(\theta_{t-1} - F'(\theta_{t-1})/L\big) \leqslant F(\theta_{t-1}) + F'(\theta_{t-1})^\top(-F'(\theta_{t-1})/L) + \frac{L}{2}\|-F'(\theta_{t-1})/L\|_2^2 \\ &= F(\theta_{t-1}) - \frac{1}{L}\|F'(\theta_{t-1})\|_2^2 + \frac{1}{2L}\|F'(\theta_{t-1})\|_2^2. \end{aligned}$$

Rearranging, we get

$$F(\theta_t) - F(\eta_*) \leqslant F(\theta_{t-1}) - F(\eta_*) - \frac{1}{2L}\|F'(\theta_{t-1})\|_2^2.$$

Using Lemma 5.1, it follows

$$F(\theta_t) - F(\eta_*) \leqslant (1 - \mu/L)(F(\theta_{t-1}) - F(\eta_*)) \leqslant \exp(-\mu/L)(F(\theta_{t-1}) - F(\eta_*)).$$

We conclude by recursion and with the definition $\kappa = L/\mu$. ■

We can make the following observations:

- As mentioned before, we necessarily have $\mu \leqslant L$; the ratio $\kappa := L/\mu$ is called the *condition number*. It is a property of the objective function, which may be hard or easy to minimize. It is not invariant by linear changes of variables $\theta \to A\theta$, where $A$ is an invertible linear map; finding a good $A$ to reduce the condition number is the main principle behind "preconditioning" techniques (see, e.g., Nocedal and Wright, 1999 for more details and the end of Section 5.2.5).
- If we only assume that the function is smooth and convex (not strongly convex), then GD with constant step-size $\gamma = 1/L$ also converges when a minimizer exists, but at a slower rate in $O(1/t)$. See Section 5.2.4 below.
- Choosing the step-size only requires an upper bound $L$ on the smoothness constant (if it is over-estimated, the convergence rate only degrades slightly).

- Writing the update $(\theta_t - \theta_{t-1})/\gamma = -F'(\theta_{t-1})$, this algorithm can be seen, under the smoothness assumption, as the discretization of the gradient flow
$$\frac{d}{dt}\eta(t) = -F'(\eta),$$
where $\eta(t\gamma) \approx \theta_t$. This analogy can lead to several insights and proof ideas (see, e.g., Scieur et al., 2017).
- For this class of functions (convex and smooth), first-order methods exist that achieve a faster rate, showing that gradient descent is not optimal. However, these improved algorithms also have drawbacks (lack of adaptivity, instability to noise, etc.). See below.

**Exercise 5.9** *Compute all constants for $\ell_2$-regularized logistic regression and for ridge regression.*

**Adaptivity.** Note that gradient descent is adaptive to strong convexity: the exact same algorithm applies to both strongly convex and convex cases, and the two bounds apply. This adaptivity is important in practice, as often, locally around the global optimum, the strong convexity constant converges to the minimal eigenvalue of the Hessian at $\eta_*$, which can be significantly larger than $\mu$ (the global constant).

**Fenchel conjugate (♦).** Given some convex function $F : \mathbb{R}^d \to \mathbb{R}$, an important tool is the Fenchel-Legendre conjugate $F^*$ defined as $F^*(\alpha) = \sup_{\theta \in \mathbb{R}^d} \alpha^\top \theta - F(\theta)$. In particular, when we allow extended-value functions (which may take the value $+\infty$), we can represent functions defined on a convex domain, and we have, under simple regularity conditions, that the conjugate of the conjugate of a convex function is the function itself. Thus, any convex function can be seen as a maximum of affine functions. Moreover, suppose the original function is not convex. In that case, the bi-conjugate is often referred to as the convex envelope and is the tightest convex lower-bound (this is often used when designing convex relaxations of non-convex problems). Moreover, using Fenchel conjugation is crucial when dealing with convex duality (which we will not cover in this chapter). See Boyd and Vandenberghe (2004) for details.

**Exercise 5.10** *Let $F$ be an $L$-smooth convex function on $\mathbb{R}^d$. Show that its Fenchel conjugate is $(1/L)$-strongly convex.*

**Exercise 5.11** *Let $F$ be an $L$-smooth convex function on $\mathbb{R}^d$, and $F^*$ its Fenchel conjugate. Show that for any $\theta, z \in \mathbb{R}^d$, we have $F(\theta)+F^*(z)-z^\top\theta \geqslant 0$, if and only if $z = F'(\theta)$ (this is the Fenchel-Young inequality). (♦) Show in addition that $F(\theta) + F^*(z) - z^\top\theta \geqslant \frac{1}{2L}\|z - F'(\theta)\|_2^2$.*

### 5.2.4 Analysis of GD for convex and smooth functions (♦)

To obtain the $1/t$ convergence rate without strong-convexity (like we got in Section 5.2.1 for quadratic functions), we will need an extra property of convex, smooth functions, sometimes called "co-coercivity". This is an instance of inequalities we need to use to circumvent the lack of closed form for iterations.

**Proposition 5.4 (co-coercivity)** *If $F$ is a convex $L$-smooth function on $\mathbb{R}^d$, then for all $\theta, \eta \in \mathbb{R}^d$, we have:*

$$\frac{1}{L}\|F'(\theta) - F'(\eta)\|_2^2 \leqslant \big[F'(\theta) - F'(\eta)\big]^\top (\theta - \eta).$$

*Moreover, we have: $F(\theta) \geqslant F(\eta) + F'(\eta)^\top(\theta - \eta) + \frac{1}{2L}\|F'(\theta) - F'(\eta)\|^2$.*

**Proof** We will show the second inequality, which implies the first one, by applying it twice with $\eta$ and $\theta$ swapped and summing them.

- Define $H(\theta) = F(\theta) - \theta^\top F'(\eta)$. The function $H : \mathbb{R}^d \to \mathbb{R}$ is convex with global minimum at $\eta$, since $H'(\theta) = F'(\theta) - F'(\eta)$, which is equal to zero for $\theta = \eta$. The function $H$ is also $L$-smooth.
- From the definition of smoothness, we get $H(\theta - \frac{1}{L}H'(\theta)) \leqslant H(\theta) + H'(\theta)^\top(-\frac{1}{L}H'(\theta)) + \frac{L}{2}\|-\frac{1}{L}H'(\theta)\|_2^2$, which is less than $H(\theta) - \frac{1}{2L}\|H'(\theta)\|_2^2$.
- This leads to $F(\eta) - \eta^\top F'(\eta) = H(\eta) \leqslant H(\theta - \frac{1}{L}H'(\theta)) \leqslant H(\theta) - \frac{1}{2L}\|H'(\theta)\|_2^2 = F(\theta) - \theta^\top F'(\eta) - \frac{1}{2L}\|F'(\theta) - F'(\eta)\|_2^2$, which leads to the desired inequality by shuffling terms.

■

We can now state the following convergence result for gradient descent with potentially no strong-convexity. Up to constants, we obtain the same rate for quadratic functions in Eq. (5.5).

**Proposition 5.5 (Convergence of GD for smooth convex functions)** *Assume that $F$ is $L$-smooth and convex, with a global minimizer $\eta_*$. Choosing $\gamma_t = 1/L$, the iterates $(\theta_t)_{t \geqslant 0}$ of GD on $F$ satisfy, for $t > 0$,*

$$F(\theta_t) - F(\eta_*) \leqslant \frac{L}{2t}\|\theta_0 - \eta_*\|_2^2.$$

**Proof** Following Bansal and Gupta (2019), the Lyapunov function that we will choose is

$$V_t(\theta_t) = t[F(\theta_t) - F(\eta_*)] + \frac{L}{2}\|\theta_t - \eta_*\|_2^2,$$

and our goal is to show that it decays along iterations (the requirement is thus weaker than for exponential convergence). We can split the difference in Lyapunov functions into three terms (each with its own color):

$$\begin{aligned} & V_t(\theta_t) - V_{t-1}(\theta_{t-1}) \\ = \;& t[F(\theta_t) - F(\theta_{t-1})] + F(\theta_{t-1}) - F(\eta_*) + \frac{L}{2}\|\theta_t - \eta_*\|_2^2 - \frac{L}{2}\|\theta_{t-1} - \eta_*\|_2^2. \end{aligned}$$

To bound it:

- We use $F(\theta_t) - F(\theta_{t-1}) \leqslant -\frac{1}{2L}\|F'(\theta_{t-1})\|_2^2$ like in the proof of Prop. 5.3.
- We use $F(\theta_{t-1}) - F(\eta_*) \leqslant F'(\theta_{t-1})^\top(\theta_{t-1} - \eta_*)$, as a consequence of convexity (function above the tangent at $\theta_{t-1}$), as in Eq. (5.7).
- We use $\frac{L}{2}\|\theta_t - \eta_*\|_2^2 - \frac{L}{2}\|\theta_{t-1} - \eta_*\|_2^2 = -L\gamma(\theta_{t-1} - \eta_*)^\top F'(\theta_{t-1}) + \frac{L\gamma^2}{2}\|F'(\theta_{t-1})\|_2^2$ by expanding the square.

This leads to, with the step-size $\gamma = 1/L$:

$$\begin{aligned} V_t(\theta_t) - V_{t-1}(\theta_{t-1}) &\leqslant t\Big[-\frac{1}{2L}\|F'(\theta_{t-1})\|_2^2\Big] + F'(\theta_{t-1})^\top(\theta_{t-1} - \eta_*) \\ &\qquad -L\gamma(\theta_{t-1} - \eta_*)^\top F'(\theta_{t-1}) + \frac{L\gamma^2}{2}\|F'(\theta_{t-1})\|_2^2 \\ &= -\frac{t-1}{2L}\|F'(\theta_{t-1})\|_2^2 \leqslant 0, \end{aligned}$$

which leads to $t[F(\theta_t) - F(\eta_*)] \leqslant V_t(\theta_t) \leqslant V_0(\theta_0) = \frac{L}{2}\|\theta_0 - \eta_*\|_2^2$,, and thus the desired bound $F(\theta_t) - F(\eta_*) \leqslant \frac{L}{2t}\|\theta_0 - \eta_*\|_2^2$. ■

The proof above is on purpose mysterious: the choice of Lyapunov function seems arbitrary at first, but all inequalities lead to nice cancellations. These proofs are sometimes hard to design. For an interesting line of work trying to automate these proofs, see `https://francisbach.com/computer-aided-analyses/`.

**Exercise 5.12** *(alternative convergence proof ♦) We consider an $L$-smooth convex function with a global minimizer $\eta_*$, and gradient descent with step-size $\gamma_t = 1/L$.*

*(a) Show that $\|\theta_t - \eta_*\|_2^2 \leqslant \|\theta_{t-1} - \eta_*\|_2^2$ for all $t \geqslant 1$.*

*(b) Show that $F(\theta_t) \leqslant F(\theta_{t-1}) - \frac{1}{2L}\|F'(\theta_{t-1})\|_2^2$ for all $t \geqslant 1$.*

*(c) Denoting $\Delta_t = F(\theta_t) - F(\eta_*)$, show that $\Delta_t \leqslant \Delta_{t-1} - \frac{1}{2L\|\theta_0 - \eta_*\|^2}\Delta_{t-1}^2$ for all $t \geqslant 1$. Conclude that $\Delta_t \leqslant \frac{2L}{t+4}\|\theta_0 - \eta_*\|^2$.*

**Early stopping for machine learning (♦♦).** An inspection of the proof of Prop. 5.5 shows that throughout the proof, a minimizer $\eta_*$ can be replaced by *any* $\eta \in \mathbb{R}^d$, leading to

$$t[F(\theta_t) - F(\eta)] + \frac{L}{2}\|\theta_t - \eta\|_2^2 \leqslant \frac{L}{2}\|\theta_0 - \eta\|_2^2. \tag{5.11}$$

When $F(\theta) = \widehat{\mathcal{R}}(\theta) = \frac{1}{n}\sum_{i=1}^n \ell(y_i, \theta^\top\varphi(x_i))$, for a smooth loss function (with constant $G_2$), for linear predictions with feature $\ell_2$-norms smaller than $R$, we have $L \leqslant G_2R^2$. Moreover, if the loss is non-negative, less than $G_0$ at zero predictions, and also Lipschitz-continuous (with constant $G_1$), such as the logistic loss, we showed in Section 4.5.4 that for any $D$,

$$\mathbb{E}\Big[\sup_{\|\theta\|_2 \leqslant D} \big|\mathcal{R}(\theta) - \widehat{\mathcal{R}}(\theta)\big|\Big] \leqslant \frac{4G_1RD}{\sqrt{n}}.$$

Then from Eq. (5.11), assuming we initialize with $\theta_0 = 0$, we get for $\eta = 0$, $\frac{G_2 R^2}{2}\|\theta_t\|_2^2 \leqslant tG_0$, leading to for any $\eta \in \mathbb{R}^d$:

$$\mathbb{E}\big[\mathcal{R}(\theta_t)\big] \leqslant \mathcal{R}(\eta) + \frac{4G_1RD}{\sqrt{n}}\Big(\|\eta\|_2 + \Big(\frac{2G_0}{G_2R^2}t\Big)^{1/2}\Big),$$

showing that if $t = o(n)$ (we do not make too many steps), the testing error is controlled. In particular, if $\theta_*$ is a minimizer of the expected risk $\mathcal{R}$, then wit $t = \sqrt{n}$, we obtain $\mathbb{E}\big[\mathcal{R}(\theta_t)\big] - \mathcal{R}(\theta_*) = O(n^{-1/4})$, which will be improved using a different analysis at the end of Section 5.3.

### 5.2.5 Beyond gradient descent (♦)

While gradient descent is the simplest algorithm with a simple analysis, there are multiple extensions that we will only briefly mention (see more details by Nesterov, 2004, 2007):

**Nesterov acceleration.** For strongly-convex functions, a simple modification of gradient descent allows for obtaining better convergence rates. The algorithm is as follows and is based on updating the following iterates:

$$\theta_t = \eta_{t-1} - \frac{1}{L}g'(\eta_{t-1}) \quad (5.12)$$

$$\eta_t = \theta_t + \frac{1-\sqrt{\mu/L}}{1+\sqrt{\mu/L}}(\theta_t - \theta_{t-1}), \quad (5.13)$$

and the convergence rate is then $F(\theta_t) - F(\eta_*) \leqslant L\|\theta_0 - \eta_*\|^2(1-\sqrt{\mu/L})^t$, which is equal to $L\|\theta_0 - \eta_*\|^2(1-1/\sqrt{\kappa})^t$, that is the characteristic time to convergence goes from $\kappa$ to $\sqrt{\kappa}$. If $\kappa$ is large (typically of order $\sqrt{n}$ or $n$ for machine learning), the gains are substantial. In practice, this leads to significant improvements. See a detailed description and many extensions by d'Aspremont et al. (2021).

For convex functions, we need the extrapolation step to depend on $t$ as follows:

$$\theta_t = \eta_{t-1} - \frac{1}{L}F'(\eta_{t-1}) \quad (5.14)$$

$$\eta_t = \theta_t + \frac{t-1}{t+2}(\theta_t - \theta_{t-1}). \quad (5.15)$$

This simple modification dates back to Nesterov in 1983 and leads to the following convergence rate $F(\theta_t) - F(\eta_*) \leqslant \frac{2L\|\theta_0-\eta_*\|^2}{(t+1)^2}$. See exercise below, and d'Aspremont et al. (2021) for more details.

Moreover, the last two rates are known to be optimal for the considered problems. For algorithms that access gradients and combine them linearly to select a new query point, it is impossible to have better dimension-independent rates. See Nesterov (2007) and Chapter 15 for more details.

**Exercise 5.13** *(♦♦) For the updates in Eq. (5.12) and Eq. (5.13), show that for* $L(\theta, \eta) = f(\theta) - f(\eta_*) + \frac{\mu}{2}\Big\|\eta - \eta_* + \frac{1+\sqrt{\mu/L}}{\sqrt{\mu/L}}(\theta - \eta)\Big\|_2^2$*, then* $L(\theta_t, \eta_t) \leqslant (1 - \sqrt{\mu/L})L(\theta_{t-1}, \eta_{t-1})$*. Show that this implies a convergence rate proportional to* $(1 - \sqrt{\mu/L})^t$*.*

**Exercise 5.14** *(♦♦) For the updates in Eq. (5.14) and Eq. (5.15), show that for* $L_t(\theta, \eta) = \left(\frac{t+1}{2}\right)^2\big[f(\theta) - f(\eta_*) + \frac{L}{2}\|\eta - \eta_* + \frac{t}{2}(\eta - \theta)\|_2^2\big]$*, then* $L_t(\theta_t, \eta_t) \leqslant L_{t-1}(\theta_{t-1}, \eta_{t-1})$*. Show that this implies a convergence rate proportional to* $\frac{1}{t^2}$*.*

**Newton method.** Given $\theta_{t-1}$, the Newton method minimizes the second-order Taylor expansion around $\theta_{t-1}$ (or, equivalently, finds a zero of $F'$ by using a first-order Taylor expansion of $F'$ around $\theta_{t-1}$):

$$F(\theta_{t-1}) + F'(\theta_{t-1})^\top(\theta - \theta_{t-1}) + \frac{1}{2}(\theta - \theta_{t-1})^\top F''(\theta_{t-1})^\top(\theta - \theta_{t-1}).$$

The gradient of this quadratic function is $\theta - \theta_{t-1} + F''(\theta_{t-1})^\top(\theta - \theta_{t-1})$, and setting it to zero leads to $\theta_t = \theta_{t-1} - F''(\theta_{t-1})^{-1}F'(\theta_{t-1})$, which is an expensive iteration, as the running-time complexity is $O(d^3)$ in general to solve the linear system. It leads to local quadratic convergence: If $\|\theta_{t-1} - \theta_*\|$ small enough, for some constant $C$, one can show $(C\|\theta_t - \theta_*\|) = (C\|\theta_{t-1} - \theta_*\|)^2$. See Boyd and Vandenberghe (2004) for more details and conditions for global convergence, in particular through the use of "self-concordance", which is a property that relates third and second-order derivatives.

⚠ The denomination "quadratic" is sometimes confusing and corresponds to a number of significant digits that doubles at each iteration.

Note that for machine learning problems, quadratic convergence may be overkill compared to the computational complexity of each iteration since cost functions are averages of $n$ terms and naturally have some uncertainty of order $O(1/\sqrt{n})$.

**Exercise 5.15** *(♦) Assume the function $F$ is $\mu$-strongly convex, twice differentiable, and such that the Hessian is Lipschitz-continuous, i.e.,* $\|f''(\theta) - f''(\eta)\|_{\text{op}} \leqslant M\|\theta - \eta\|_2$*. Using the Taylor formula with integral remainder, show that for the iterates of Newton's method,* $\|\nabla F(\theta_t)\|_2 \leqslant \frac{M}{2\mu^2}\|\nabla F(\theta_{t-1})\|_2^2$*. Show that this implies local quadratic convergence.*

**Proximal gradient descent (♦).** Many optimization problems are said "composite", that is, the objective function $F$ is the sum of a smooth function $G$ and a non-smooth function $H$ (such as a norm). It turns out that a simple modification of gradient descent allows us to benefit from the fast convergence rates of smooth optimization (compared to the slower rates for non-smooth optimization that we would obtain from the subgradient method in the next section).

For this, we need to first see gradient descent as a *proximal method*. Indeed, one may see the iteration $\theta_t = \theta_{t-1} - \frac{1}{L}G'(\theta_{t-1})$, as

$$\theta_t = \arg\min_{\theta \in \mathbb{R}^d} \; G(\theta_{t-1}) + (\theta - \theta_{t-1})^\top G'(\theta_{t-1}) + \frac{L}{2}\|\theta - \theta_{t-1}\|_2^2,$$

where, for a $L$-smooth function $G$, the objective function above is an upper-bound of $G(\theta)$ which is tight at $\theta_{t-1}$ (see Eq. (5.10)).

While this reformulation does not bring much for gradient descent, we can extend this to the composite problem and consider the following iteration, where $H$ is left as is,

$$\theta_t = \arg\min_{\theta\in\mathbb{R}^d} \ G(\theta_{t-1}) + (\theta-\theta_{t-1})^\top G'(\theta_{t-1}) + \frac{L}{2}\|\theta-\theta_{t-1}\|_2^2 + H(\theta).$$

It turns out that the convergence rates for $G+H$ are the same as smooth optimization, with potential acceleration (Nesterov, 2007; Beck and Teboulle, 2009).

The crux is to be able to compute the step above, that is, minimize with respect to $\theta$ functions of the form $\frac{L}{2}\|\theta-\eta\|_2^2 + H(\theta)$. When $H$ is the indicator function of a convex set (which is equal to 0 inside the set, and $+\infty$ otherwise), we get projected gradient descent. When $H$ is the $\ell_1$-norm, that is, $H = \lambda\|\cdot\|_1$, this can be shown to be a soft-thresholding step, as for each coordinate $\theta_i = (|\eta_i| - \lambda/L)_+ \frac{\eta_i}{|\eta_i|}$ (proof left as an exercise). See applications to model selection and sparsity-inducing norms in Chapter 8.

**Pre-conditioning (♦).** The convergence rate of gradient descent depends crucially on the condition number $\kappa$, which is not invariant by linear rescaling of the problem. That is, if we (equivalently) aim to minimize $G(\tilde\theta) = F(A\tilde\theta)$ for some invertible matrix $A\in\mathbb{R}^{d\times d}$ and a twice differentiable function $F$, the gradient of $G$ is $G'(\tilde\theta) = A^\top F'(A\tilde\theta)$, and thus gradient descent on $G$ can be written $\tilde\theta_t = \tilde\theta_{t-1} - \gamma G'(\tilde\theta) = \tilde\theta_{t-1} - \gamma A^\top F'(A\tilde\theta_{t-1})$, which can be rewritten as $\theta_t = \theta_{t-1} - \gamma AA^\top F'(\theta_{t-1})$ with the change of variable $\theta = A\tilde\theta$. This is thus equivalent to pre-multiplying the gradient of $F$ by the positive definite matrix $AA^\top$.

This will be advantageous when the condition number of $G$ is smaller than that of $F$. For example, for a quadratic function $F$ with constant Hessian matrix $H\in\mathbb{R}^{d\times d}$, taking $A$ as an inverse square root of $H$ leads to the minimal possible value of the condition number, and thus the pre-conditioned gradient iteration (here equal to the Newton step) converges in one iteration. Such a value of $A$ optimizes the condition number but is not computationally efficient, and various conditioners can be used in practice, based on diagonal approximations of the Hessian, random projections (Martinsson and Tropp, 2020) or incomplete Cholesky factorizations (Golub and Loan, 1996). Such preconditioning is also useful in non-smooth situations (see Section 5.4.2 in the context of SGD).

### 5.2.6 Non-convex objective functions (♦)

For smooth, potentially non-convex objective functions, the best one can hope for is to converge to a stationary point $\theta$ such that $F'(\theta) = 0$. The proof below provides the weaker result that at least one iterate has a small gradient. Indeed, using the same Taylor expansion as the convex case (which is still valid), we get, using the $L$-smoothness of $F$:

$$F(\theta_t) \ \leqslant\ F(\theta_{t-1}) - \frac{1}{2L}\|F'(\theta_{t-1})\|_2^2,$$

leading to, summing the inequalities above for all iterations between 1 and $t$:

$$\frac{1}{2Lt}\sum_{s=1}^{t}\|F'(\theta_{s-1})\|_2^2 \leqslant \frac{F(\theta_0)-F(\eta_*)}{t}.$$

Thus, there is one $s$ in $\{0,\dots,t-1\}$ for which $\|F'(\theta_s)\|_2^2 \leqslant O(1/t)$. Without further assumptions, this does not imply that any of the iterates is close to a stationary point.

## 5.3 Gradient methods on non-smooth problems

We now relax our assumptions and only require Lipschitz continuity in addition to convexity. The rates will be slower, but extending stochastic gradients will be easier.

**Definition 5.4 (Lipschitz-continuous function)** *A function $F:\mathbb{R}^d \to \mathbb{R}$ is said $B$-Lipschitz-continuous if and only if*

$$|F(\eta)-F(\theta)| \leqslant B\|\eta-\theta\|_2, \qquad \forall \theta,\eta \in \mathbb{R}^d.$$

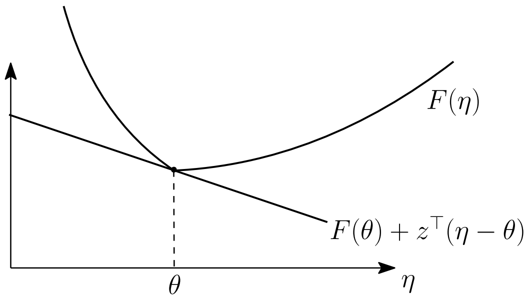
This setting is usually referred to as *non-smooth* optimization.

**Exercise 5.16** *Show that if $F$ is differentiable, $B$-Lipschitz-continuity is equivalent to the assumption $\|F'(\theta)\|_2 \leqslant B$, $\forall \theta \in \mathbb{R}^d$.*

**From gradients to subgradients.** We can apply non-smooth optimization to objective functions that are not differentiable (such as the hinge loss from Section 4.1.2). For convex Lipschitz-continuous objectives, one can show that the function is almost everywhere differentiable (see, e.g., Nekvinda and Zajíček, 1988). In points where it is not, one can define the set of slopes of lower-bounding tangents as the *subdifferential* and any element of it as a *subgradient*. That is, we can define the subdifferential as (see illustration below):

$$\partial F(\theta) = \big\{z \in \mathbb{R}^d,\ \forall \eta \in \mathbb{R}^d,\ F(\eta) \geqslant F(\theta) + z^\top(\eta-\theta)\big\}.$$

For a convex function defined on $\mathbb{R}^d$, the subdifferential happens to be a non-empty convex set at all points $\theta$. Moreover, when $F$ is differentiable with gradient $F'(\theta)$, the subdifferential is reduced to a point, that is, $\partial F(\theta) = \{F'(\theta)\}$. For example, the absolute value $\theta \mapsto |\theta|$ has a subdifferential equal to $[-1,1]$ at zero. See more details by Rockafellar (1997).



The gradient descent iteration is then meant as using any subgradient $z \in \partial F(\theta_{t-1})$ instead of $F'(\theta_{t-1})$, for which we will only need that the function is above the tangent defined by this subgradient. The method is then referred to as the subgradient method (it is not a descent method anymore, that is, the function values may go up occasionally).

**Exercise 5.17** *Compute the subdifferential of $\theta \mapsto |\theta|$ and $\theta \mapsto (1-y\theta^\top x)_+$, for the label $y \in \{-1, 1\}$ and the input $x \in \mathbb{R}^d$.*

**Convergence rate of the subgradient method.** We can prove convergence of the gradient descent algorithm, now with a decaying step-size and a slower rate than for smooth functions.

⚠ Like for stochastic gradient descent in the next section, and as opposed to gradient descent for smooth functions in the previous section, the objective function for the subgradient method for non-smooth functions may not go down at every iteration.

**Proposition 5.6 (Convergence of the subgradient method)** *Assume that $F$ is convex, $B$-Lipschitz-continuous, and admits a minimizer $\eta_*$ that satisfies $\|\eta_* - \theta_0\|_2 \leqslant D$. By choosing $\gamma_t = \frac{D}{B\sqrt{t}}$ then the iterates $(\theta_t)_{t \geqslant 0}$ of GD on $F$ satisfy*

$$\min_{0 \leqslant s \leqslant t-1} F(\theta_s) - F(\eta_*) \leqslant DB \frac{2 + \log(t)}{2\sqrt{t}}. \tag{5.16}$$

**Proof** We look at how $\theta_t$ approaches $\eta_*$, that is, we try to use $\|\theta_t - \eta_*\|_2^2$ as a Lyapunov function. We have:

$$\begin{aligned} \|\theta_t - \eta_*\|_2^2 &= \|\theta_{t-1} - \gamma_t F'(\theta_{t-1}) - \eta_*\|_2^2 \\ &= \|\theta_{t-1} - \eta_*\|_2^2 - 2\gamma_t F'(\theta_{t-1})^\top(\theta_{t-1} - \eta_*) + \gamma_t^2 \|F'(\theta_{t-1})\|_2^2. \end{aligned}$$

Combining this with the convexity inequality $F(\theta_{t-1}) - F(\eta_*) \leqslant F'(\theta_{t-1})^\top(\theta_{t-1} - \eta_*)$ from Eq. (5.7), using the boundedness of the gradients (that is, $\|F'(\theta_{t-1})\|_2^2 \leqslant B^2$), it follows:

$$\|\theta_t - \eta_*\|_2^2 \leqslant \|\theta_{t-1} - \eta_*\|_2^2 - 2\gamma_t[F(\theta_{t-1}) - F(\eta_*)] + \gamma_t^2 B^2.$$

We are in a situation where the Lyapunov function $\theta \mapsto \|\theta - \eta_*\|_2^2$ is not decreasing along iterations because of the term $\gamma_t^2 B^2$ above. It is then classical to isolate the negative term $-2\gamma_t[F(\theta_{t-1}) - F(\eta_*)]$ and sum inequalities. Thus, by isolating the distance to optimum in function values, we get:

$$\gamma_t(F(\theta_{t-1}) - F(\eta_*)) \leqslant \frac{1}{2}\Big(\|\theta_{t-1} - \eta_*\|_2^2 - \|\theta_t - \eta_*\|_2^2\Big) + \frac{1}{2}\gamma_t^2 B^2. \tag{5.17}$$

It is sufficient to sum these inequalities to get (in fact, for any $\eta_* \in \mathbb{R}^d$ and not only the minimizer),

$$\frac{1}{\sum_{s=1}^t \gamma_s} \sum_{s=1}^t \gamma_s \left(F(\theta_{s-1}) - F(\eta_*)\right) \leqslant \frac{\|\theta_0 - \eta_*\|_2^2}{2\sum_{s=1}^t \gamma_s} + B^2 \frac{\sum_{s=1}^t \gamma_s^2}{2\sum_{s=1}^t \gamma_s}.$$

As a weighted average, the left-hand side is larger than $\min_{0\leqslant s\leqslant t-1}(F(\theta_s) - F(\eta_*))$, and also larger than $F(\bar{\theta}_t) - F(\eta_*)$ where $\bar{\theta}_t = (\sum_{s=1}^t \gamma_s \theta_{s-1})/(\sum_{s=1}^t \gamma_s)$ by Jensen's inequality.

The upper bound goes to 0 if $\sum_{s=1}^t \gamma_s$ goes to $\infty$ (to forget the initial condition, sometimes called the "bias") and $\gamma_t \to 0$ (to decrease the "variance" term). Let us choose $\gamma_s = \tau/\sqrt{s}$ for some $\tau > 0$. By using the series-integral comparisons below, we get the bound

$$\min_{0\leqslant s\leqslant t-1}(F(\theta_s) - F(\eta_*)) \leqslant \frac{1}{2\sqrt{t}}\Big(\frac{D^2}{\tau} + \tau B^2(1 + \log(t))\Big).$$

We choose $\tau = D/B$ (which is suggested by optimizing the previous bound when $\log(t) = 0$), which leads to the result. In the proof, we used the following series-integral comparisons for decreasing functions:

$$\sum_{s=1}^t \frac{1}{\sqrt{s}} \geqslant \sum_{s=1}^t \frac{1}{\sqrt{t}} = \sqrt{t},$$

and $\sum_{s=1}^t \frac{1}{s} \leqslant 1 + \sum_{s=2}^t \frac{1}{s} \leqslant 1 + \int_1^t \frac{ds}{s} = 1 + \log(t)$. ■

The proof scheme above is very flexible. It can be extended in the following directions:

- There is no need to know in advance an upper-bound $D$ on the distance to optimum; we then get, with an arbitrary $D$, with the same step-size $\gamma_t = \frac{D}{B\sqrt{t}}$ a rate of the form $\frac{BD}{2\sqrt{t}}\big(\frac{\|\theta_0-\eta_*\|_2^2}{D^2} + (1 + \log(t))\big)$. Moreover, a slightly modified version of the subgradient method removes the need to know the Lipschitz constant. See the exercise below.

  **Exercise 5.18** *We consider the iteration $\theta_t = \theta_{t-1} - \frac{\gamma'_t}{\|F'(\theta_{t-1})\|_2} F'(\theta_{t-1})$. Show that with the step-size $\gamma'_t = D/\sqrt{t}$, we get the guarantee $\min_{0\leqslant s\leqslant t-1} F(\theta_s) - F(\eta_*) \leqslant DB\frac{2+\log(t)}{2\sqrt{t}}$.*

- The algorithm applies to constrained minimization over a convex set by inserting a projection step at each iteration (the proof, which uses the contractivity of orthogonal projections, is essentially the same; see the exercise below).

  **Exercise 5.19** *Let $K \subset \mathbb{R}^d$ be a convex closed set, and denote by $\Pi_K(\theta)$ he orthogonal projection of $\theta$ onto $K$, defined as $\Pi_K(\theta) = \arg\min_{\eta\in K} \|\eta - \theta\|_2^2$.Show that the function $\Pi_K$ is contractive, that is, for all $\theta, \eta \in \mathbb{R}^d$, $\|\Pi_K(\theta) - \Pi_K(\eta)\|_2 \leqslant \|\theta - \eta\|_2$. For the algorithm $\theta_t = \Pi_K(\theta_{t-1} - \gamma_t F'(\theta_{t-1}))$, and $\eta_*$ a minimizer of $F$ on $K$, show that the guarantee of Prop. 5.6 still holds.*

- The algorithm applies to non-differentiable convex and Lipschitz objective functions (using sub-gradients, i.e., any vector satisfying Eq. (5.6) in place of $F'(\theta_t)$).

- The algorithm can be applied to "non-Euclidean geometries", where we consider bounds on the iterates or the gradient with different quantities, such as Bregman divergences. This can be done using the "mirror descent" framework, and for instance, can be applied to obtain multiplicative updates (see, e.g., Juditsky and Nemirovski, 2011a,b; Bubeck, 2015). See more details in the online stochastic case in Section 11.1.3.

**Exercise 5.20** *(♦) Let $F : \mathbb{R}^d \to \mathbb{R}$ be a differentiable function, and $\psi : \mathbb{R}^d \to \mathbb{R}$ a strictly convex function:*

*(a) Show that the minimizer of $F(\theta) + F'(\theta)^\top(\eta - \theta) + \frac{1}{2\gamma}\|\eta - \theta\|_2^2$ is equal to $\eta = \theta - \gamma F'(\theta)$.*

*(b) Show that the Bregman divergence $D_\psi(\eta, \theta)$ defined as $D_\psi(\eta, \theta) = \psi(\eta) - \psi(\theta) - \psi'(\theta)^\top(\eta - \theta)$ is non-negative, and equal to zero if and only if $\eta = \theta$.*

*(c) Show that the minimizer of $F(\theta) + F'(\theta)^\top(\eta - \theta) + \frac{1}{\gamma}D_\psi(\eta, \theta)$ satisfies $\psi'(\eta) = \psi'(\theta) - \gamma F'(\theta)$. Show that the same conclusion holds if $\psi$ is only defined on an open convex set $K \subset \mathbb{R}^d$, and so that the gradient $\psi'$ is a bijection from $K$ to $\mathbb{R}^d$.*

*(d) Apply to $\psi(\theta) = \sum_{i=1}^d \theta_i \log \theta_i$.*

- Often the uniformly averaged iterate is used, as $\frac{1}{t}\sum_{s=0}^{t-1}\theta_s$. Convergence rates (without the $\log t$ factor) can be obtained with a slightly more involved proof using the Abel summation formula (see also Section 11.1.1).

**Exercise 5.21** *(♦) We consider the same assumptions as Exercise 5.19 and the same algorithm with orthogonal projections. With $D$ the diameter of $K$, show that for the average iterate $\bar\theta_t = \frac{1}{t}\sum_{s=0}^{t-1}\theta_s$, we have: $F(\bar\theta_t) - F(\theta_*) \leqslant \frac{3BD}{2\sqrt{t}}$.*

- The algorithm with the decaying step-size $\gamma_t$ is an "anytime" algorithm; that is, it can be stopped at any time $t$, and the bound in Eq. (5.16) then applies. Computations are often easier when considering a constant step-size $\gamma$ that depends on the number of iterations $T$ that the user wishes to perform, $T$ being usually referred to as the "horizon". Starting from Eq. (5.17), we get the bound:

$$\frac{1}{T}\sum_{t=1}^{T} F(\theta_{t-1}) - F(\theta_*) \leqslant \frac{D^2}{2\gamma T} + \frac{\gamma B^2}{2}, \tag{5.18}$$

  where the optimal $\gamma$ can be obtain as $\gamma = \frac{D}{B\sqrt{T}}$ and an optimized rate of $\frac{DB}{\sqrt{T}}$. We gain on the logarithmic factor, but we no longer have an anytime algorithm (since the bound only applies at time $T$). This applies as well to SGD in Section 5.4. In these situations, a "doubling trick" can be used, leading to an anytime algorithm with the same guarantee but undesired practical behavior as the algorithm makes substantial changes at each iteration, which is a power of two (see the exercise below).

- Stochastic gradients can be used, as presented below (one interpretation is that the subgradient method is so slow that it is robust to noisy gradients).

**Exercise 5.22 (doubling trick for subgradient method)** *We consider an algorithm that successively applies the SGD iteration with step-size* $\gamma = \frac{D}{B\sqrt{2^k}}$ *during* $2^k$ *iterations, for* $k = 0, 1, \dots$. *Show that after* $t$ *subgradient iterations, the observed best value of* $F$ *is less than a constant times* $DB/\sqrt{t}$.

**Exercise 5.23** *Compute all constants for* $\ell_2$*-regularized logistic regression and the support vector machine with linear predictors (Section 4.1).*

**Machine learning with linear predictions and Lipschitz-continuous losses.** For specialized machine learning problems, we can now close the loop on the discussion outlined in Section 5.1 regarding the need to take into account the optimization error on top of the estimation error. For convex Lipschitz-continuous losses (with constant $G$) such as the logistic loss or the hinge loss, for linear predictions with feature $\ell_2$-norms smaller than $R$, a parameter bounded in $\ell_2$-norm by $D$, we showed in Section 4.5.4 that the estimation error was upper-bounded by a constant times $\frac{GRD}{\sqrt{n}}$. The optimization error after $t$ iterations of the subgradient method is upper-bounded by a constant times $\frac{GRD}{\sqrt{t}}$, since the Lipschitz constant of the objective function is $B \leqslant GR$.

Adding these two bounds, there is no need to have the number of iterations $t$ larger than the number of observations $n$. However, since each gradient computation requires $n$ gradient computations for the individual loss functions associated with a single data point, the total number of such gradient computations is $tn \approx n^2$, which is not scalable when $n$ is large. We now show how stochastic gradient descent can turn this number to $n$ with the same upper-bound on the generalization performance.

## 5.4 Convergence rate of stochastic gradient descent (SGD)

For machine learning problems, where $F(\theta) = \frac{1}{n}\sum_{i=1}^n \ell(y_i, f_\theta(x_i)) + \Omega(\theta)$, at each iteration, the gradient descent algorithm requires computing a "full" gradient $F'(\theta_{t-1})$, which could be costly as it requires accessing the entire data set (all $n$ pairs of observations). An alternative is to instead only compute *unbiased* stochastic estimations of the gradient $g_t(\theta_{t-1})$, i.e., such that

$$\mathbb{E}[g_t(\theta_{t-1})|\theta_{t-1}] = F'(\theta_{t-1}), \tag{5.19}$$

which could be much faster to compute, in particular by accessing fewer observations.

⚠ Note that we need to condition over $\theta_{t-1}$ because $\theta_{t-1}$ encapsulates all the randomness due to past iterations, and we only require "fresh" randomness at time $t$.

⚠ Somewhat surprisingly, this unbiasedness does *not* need to be coupled with a vanishing variance: while there are always errors in the gradient, the use of a decreasing step-size will ensure convergence. If the noise in the gradient is not unbiased, then we only get convergence if the noise magnitudes go to zero (see, e.g., d'Aspremont, 2008; Schmidt et al., 2011 and references therein).

This leads to the following algorithm.

**Algorithm 5.2 (Stochastic gradient descent (SGD))** *Choose a step-size sequence* $(\gamma_t)_{t \geqslant 0}$*, pick* $\theta_0 \in \mathbb{R}^d$ *and for* $t \geqslant 1$*, let*

$$\theta_t = \theta_{t-1} - \gamma_t g_t(\theta_{t-1}),$$

*where* $g_t(\theta_{t-1})$ *satisfies Eq. (5.19).*

**SGD in machine learning.** There are two ways to use SGD for supervised machine learning:

(1) **Empirical risk minimization**: If $F(\theta) = \frac{1}{n}\sum_{i=1}^{n} \ell(y_i, f_\theta(x_i))$ then at iteration $t$ we can choose uniformly at random $i(t) \in \{1, \dots, n\}$ and define $g_t$ as the gradient of $\theta \mapsto \ell(y_{i(t)}, f_\theta(x_{i(t)}))$. Here, the randomness comes from the random choice of indices.

There exist "mini-batch" variants where, at each iteration, the gradient is averaged over a random subset of the indices (we then reduce the variance of the gradient estimate, but we use more gradients, and thus the running time increases, see Exercise 5.25). We then converge to a *minimizer* $\eta_*$ *of the empirical risk.*

Note here that since we sample *with replacement*, a given function will be selected several times, even within $n$ iterations. Sampling without replacement can also be studied, but its analysis is more involved (see, e.g., Nagaraj et al., 2019, and references therein).

(2) **Expected risk minimization**: If $F(\theta) = \mathbb{E}[\ell(y, f_\theta(x))]$ is the expected (non-observable) risk, then at iteration $t$ we can take a fresh sample $(x_t, y_t)$ and define $g_t$ as the gradient of $\theta \mapsto \ell(y_t, f_\theta(x_t))$, for which, if we swap the orders of expectation and differentiation, we get the unbiasedness. Note here that to preserve the unbiasedness, only a single pass is allowed (otherwise, this would create dependencies that would break it) and that the randomness comes from the observations $(x_t, y_t)$ themselves.

Here, we *directly minimize the (generalization) risk.* The counterpart is that if we only have $n$ samples, then we can only run $n$ SGD iterations, and when $n$ grows, the iterates will converge to a *minimizer* $\theta_*$ *of the expected risk.*

Note that in practice, multiple passes over the data (that is, using each observation multiple times) lead to better performance. To avoid overfitting, either a regularization term is added to the empirical risk, or the SGD algorithm is stopped before its convergence (and typically when some validation risk stops decreasing), which is referred to as regularization by "early stopping".

We can study the two situations above using the latter one by considering the empirical risk as the expectation with respect to the empirical distribution of the data (and we thus use the notation $\theta_*$ for the global minimizer).

Stochastic gradient descent is not a descent method: the function values often go up, but they go down "on average". See, for example, an illustration in Figure 5.2.

Under the same usual assumptions on the objective functions, we now study SGD with the following extra assumptions:

- (H-1) unbiased gradient: $\mathbb{E}[g_t(\theta_{t-1})|\theta_{t-1}] = F'(\theta_{t-1})$, $\forall t \geqslant 1$,
- (H-2) bounded gradient: $\|g_t(\theta_{t-1})\|_2^2 \leqslant B^2$ almost surely, $\forall t \geqslant 1$.

Assumption (H-2) could be replaced by other regularity conditions (e.g., Lipschitz-continuous gradients). Assumption (H-1) is crucial and is often obtained by considering independent gradient functions $g_t$, for which we have $\mathbb{E}[g_t(\cdot)] = F'(\cdot)$. See Exercise 5.26 for SGD for smooth functions.

**Proposition 5.7 (Convergence of SGD)** *Assume that $F$ is convex, $B$-Lipschitz and admits a minimizer $\theta_*$ that satisfies $\|\theta_* - \theta_0\|_2 \leqslant D$. Assume that the stochastic gradients satisfy (H-1) and (H-2). Then, choosing $\gamma_t = (D/B)/\sqrt{t}$, the iterates $(\theta_t)_{t\geqslant 0}$ of SGD on $F$ satisfy*

$$\mathbb{E}\big[F(\bar{\theta}_t) - F(\theta_*)\big] \leqslant DB\frac{2 + \log(t)}{2\sqrt{t}}.$$

*where $\bar{\theta}_t = (\sum_{s=1}^t \gamma_s \theta_{s-1})/(\sum_{s=1}^t \gamma_s)$.*

We state our bound in terms of the average iterates because the cost of finding the best iterate could be higher than that of evaluating a stochastic gradient (since we cannot compute $F$ in general).

**Proof** We follow essentially the same proof as in the deterministic case, adding some expectations at well-chosen places. We have:

$$\begin{aligned}\mathbb{E}\big[\|\theta_t - \theta_*\|_2^2\big] &= \mathbb{E}\big[\|\theta_{t-1} - \gamma_t g_t(\theta_{t-1}) - \theta_*\|_2^2\big] \\ &= \mathbb{E}\big[\|\theta_{t-1} - \theta_*\|_2^2\big] - 2\gamma_t \mathbb{E}\big[g_t(\theta_{t-1})^\top(\theta_{t-1} - \theta_*)\big] + \gamma_t^2 \mathbb{E}\big[\|g_t(\theta_{t-1})\|_2^2\big].\end{aligned}$$

We can then compute the expectation of the middle term as:

$$\begin{aligned}\mathbb{E}\big[g_t(\theta_{t-1})^\top(\theta_{t-1} - \theta_*)\big] &= \mathbb{E}\big[\mathbb{E}\big[g_t(\theta_{t-1})^\top(\theta_{t-1} - \theta_*)\big|\theta_{t-1}\big]\big] \\ &= \mathbb{E}\big[\mathbb{E}\big[g_t(\theta_{t-1})\big|\theta_{t-1}\big]^\top(\theta_{t-1} - \theta_*)\big] = \mathbb{E}\big[F'(\theta_{t-1})^\top(\theta_{t-1} - \theta_*)\big],\end{aligned}$$

where we have crucially used the unbiasedness assumption (H-1). This leads to

$$\mathbb{E}\big[\|\theta_t - \theta_*\|_2^2\big] \leqslant \mathbb{E}\big[\|\theta_{t-1} - \theta_*\|_2^2\big] - 2\gamma_t \mathbb{E}\big[F'(\theta_{t-1})^\top(\theta_{t-1} - \theta_*)\big] + \gamma_t^2 B^2.$$

Thus, combining with the convexity inequality $F(\theta_{t-1}) - F(\theta_*) \leqslant F'(\theta_{t-1})^\top(\theta_{t-1} - \theta_*)$ from Eq. (5.7), we get

$$\gamma_t \mathbb{E}[F(\theta_{t-1}) - F(\theta_*)] \leqslant \frac{1}{2}\Big(\mathbb{E}\big[\|\theta_{t-1} - \theta_*\|_2^2\big] - \mathbb{E}\big[\|\theta_t - \theta_*\|_2^2\big]\Big) + \frac{1}{2}\gamma_t^2 B^2. \qquad (5.20)$$

Except for the expectations, this is the same bound as Eq. (5.17), so we can conclude as in the proof of Prop. 5.6. ■

We can make the following observations:

- Averaging of iterates is often performed after a certain number of iterations (e.g., one pass over the data when doing multiple passes): having such a "burn-in" period speeds up the algorithms by forgetting initial conditions faster.
- Many authors consider the projected version of the algorithm where after the gradient step, we orthogonally project onto the ball of radius $D$ and center $\theta_0$. The bound is then exactly the same.
- Like for the subgradient method in Eq. (5.18), we can consider a constant step-size $\gamma$, to obtain

  $$\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}\big[F(\theta_{t-1})\big]-F(\theta_*)\leqslant\frac{D^2}{2\gamma T}+\frac{\gamma B^2}{2},$$

  from which we get, $\mathbb{E}\big[F(\bar\theta_t)\big]-F(\theta_*)\leqslant\frac{D^2}{2\gamma T}+\frac{\gamma B^2}{2}=\frac{DB}{\sqrt{T}}$, for the specific choice $\gamma=D/(B\sqrt{T})$ (that depends on the horizon $T$, and for the uniformly average iterate.
- The result that we obtain, when applied to single pass SGD, is a generalization bound; that is, after the $n$ iterations, we have an excess risk proportional to $1/\sqrt{n}$, corresponding to the excess risk compared to the best predictor $f_\theta$.

  This is to be compared to using results from Chapter 4 (uniform deviation bounds) and non-stochastic gradient descent. It turns out that the estimation error due to having $n$ observations is exactly the same as the generalization bound obtained by SGD (see Section 4.5.4 in Chapter 4). Still, we need to add on top of the optimization error proportional to $1/\sqrt{t}$ (with the same constants). The bounds match if $t=n$, that is, we run $n$ iterations of gradient descent on the empirical risk. This leads to a running time complexity of $O(tnd)=O(n^2d)$ instead of $O(nd)$ using SGD, hence the strong gains in using SGD.

  ⚠ We are still comparing upper bounds.
- The bound in $O(BD/\sqrt{t})$ is optimal for this class of problem. That is, as shown for example by Agarwal et al. (2009), among all algorithms that can query stochastic gradients, having a better convergence rate (up to some constants) is impossible. See Section 15.3 for a detailed proof.
- As opposed to the deterministic case, the use of smoothness does not lead to significantly better results (see Exercise 5.26).
- An inspection of the proof shows that we can replace the almost sure bounds $\|g_t(\theta_{t-1})\|_2^2\leqslant B^2$ by bounds in expectation $\mathbb{E}\big[\|g_t(\theta_{t-1})\|_2^2\big]\leqslant B^2$. For machine learning problems with linear predictions with feature $\ell_2$-norm bounded by $R$ and a $G$-Lipschitz-continuous loss, the gradient $g_t(\theta_{t-1})$ is the gradient of the function $\theta\mapsto\ell(y_t,\varphi(x_t)^\top\theta)$ taken at $\theta_{t-1}$, and thus its squared norm is less than

$G^2 \cdot \|\varphi(x_t)\|_2^2$. An almost sure bound is therefore $G^2R^2$, while a bound in expectation is $G^2 \cdot \mathbb{E}\big[\|\varphi(x_t)\|_2^2\big]$, which is stronger.

- We can obtain a result in high probability, using an extension of Hoeffding's inequality to "differences of martingales", as shown in the exercise below.

**Exercise 5.24 (high-probability bound for SGD (♦))** *Using the same assumptions and notations as in Prop. 5.7, we consider the projected iteration: $\theta_t = \Pi_D(\theta_{t-1} - \gamma_t g_t)$, where $\Pi_D$ is the orthogonal projection on the $\ell_2$-ball of center $0$ and radius $D$. Denoting $z_t = -\gamma_t(\theta_{t-1} - \theta_*)^\top[g_t - F'(\theta_{t-1})]$, show that $\mathbb{E}[z_t|\mathcal{F}_{t-1}] = 0$ and $|z_t| \leqslant 4\gamma_t BD$ almost surely, and that*

$$\gamma_t[F(\theta_{t-1}) - F(\theta_*)] \leqslant \frac{1}{2}\Big(\mathbb{E}\big[\|\theta_{t-1} - \theta_*\|_2^2\big] - \mathbb{E}\big[\|\theta_t - \theta_*\|_2^2\big]\Big) + \frac{1}{2}\gamma_t^2 B^2 + z_t.$$

*Show that with probability at least $1-\delta$, then, for the weighted average $\bar\theta_t$ defined in Prop. 5.7, for any step-sizes $\gamma_t$:*

$$F(\bar\theta_t) \leqslant \frac{2D^2}{\sum_{s=1}^t \gamma_s} + B^2 \frac{\sum_{s=1}^t \gamma_s^2}{2\sum_{s=1}^t \gamma_s} + 4BD\frac{\big(\sum_{s=1}^t \gamma_s^2\big)^{1/2}}{\sum_{s=1}^t \gamma_s}\sqrt{2\log\frac{1}{\delta}},$$

*and that for a constant-step size, $\gamma_t = \gamma$, $F(\bar\theta_t) \leqslant \frac{2D^2}{\gamma T} + \frac{\gamma B^2}{2} + \frac{4DB}{\sqrt{t}}\sqrt{2\log\frac{1}{\delta}}$.*

**SGD or gradient descent on the empirical risk?** As seen above, the number of iterations to reach a given precision will be larger for stochastic gradient descent than for smooth deterministic gradient descent, but with a complexity that is typically $n$ times faster. Thus, for high precision, that is, low values of $F(\theta) - F(\eta_*)$ (which is not needed for machine learning), the number of iterations of SGD may become prohibitively large, and deterministic full gradient descent could be preferred. However, for low precision and large $n$, SGD is the method of choice (see also recent improvements in Section 5.4.4).

In particular, for the linear prediction case described at the end of Section 5.3, we obtain the exact same rate in Prop. 5.7 as for non-stochastic gradient descent on the empirical risk. If sampling from the $n$ observations with replacement, after $t = n$ steps, the sum of optimization error and optimization error is of the same order $O(\frac{GRD}{\sqrt{n}})$, with now only $n$ accesses to individual loss gradients (instead of $n^2$ with batch methods, thus, with a big improvement). Moreover, with *a single pass over the data*, Prop. 5.7 is *directly* a generalization performance result with the same rate.

**Exercise 5.25** *We consider the mini-batch version of SGD, where at every iteration, we replace $g_t(\theta_{t-1})$ by the average of $m$ independent samples of stochastic gradients at $\theta_{t-1}$. Extend the convergence result of Prop. 5.7.*

**Exercise 5.26** *(♦) We consider independent and identically distributed convex $L$-smooth random functions $f_t : \mathbb{R}^d \to \mathbb{R}$, $t \geqslant 1$, with expectation $F : \mathbb{R}^d \to \mathbb{R}$, that has a minimizer $\theta_* \in \mathbb{R}^d$. We consider the SGD recursion $\theta_t = \theta_{t-1} - \gamma_t f_t'(\theta_{t-1})$, with $\gamma_t$ a deterministic* *step-size sequence. Using co-coercivity (Prop. 5.4), show that*

$$\mathbb{E}\big[\|\theta_t-\theta_*\|_2^2\big] \leqslant \mathbb{E}\big[\|\theta_{t-1}-\theta_*\|_2^2\big]-2\gamma_t(1-\gamma_t L)\mathbb{E}\big[F'(\theta_{t-1})^\top(\theta_{t-1}-\theta_*)\big]+2\gamma_t^2\mathbb{E}\big[\|f_t'(\theta_*)\|_2^2\big].$$

*Extend the proof of Prop. 5.7 to obtain an explicit rate in $O(1/\sqrt{t})$.*

**Exercise 5.27 (non-uniform sampling (♦))** *We consider functions $F : \mathbb{R}^d \times \mathcal{Z} \to \mathbb{R}$, which are convex with respect to the first variable, with a subgradient $F'(\theta, z)$ with respect to the first variable which is bounded in $\ell_2$-norm by a constant $B(z)$ that depends on $z$. We consider a distribution $p$ on $\mathcal{Z}$, and we aim to minimize $\mathbb{E}_{z\sim p}F(\theta, z)$, but we sample from a distribution $q$, with density $dq/dp(z)$ with respect to $p$ to get independent and identically distributed random $z_t$, $t \geqslant 1$. We consider the recursion $\theta_t = \theta_{t-1} - \frac{\gamma}{dq/dp(z_t)}F'(\theta_{t-1}, z_t)$. Provide a convergence rate for this algorithm and show how a good choice of $q$ leads to significant improvements over the choice $q = p$ when $B(z)$ is far from uniform in $z$. Apply this result to the support vector machine when applying SGD to the empirical risk.*

### 5.4.1 Strongly convex problems (♦)

We consider the regularized problem $G(\theta) = F(\theta) + \frac{\mu}{2}\|\theta\|_2^2$, with the same assumption as above, and started at $\theta_0 = 0$. The SGD iteration is then, with $g_t(\theta_{t-1})$ a stochastic (sub)gradient of $F$ at $\theta_{t-1}$:

$$\theta_t = \theta_{t-1} - \gamma_t\big[g_t(\theta_{t-1}) + \mu\theta_{t-1}\big]. \tag{5.21}$$

We then have an improved convergence rate in $O(1/t)$ with a different decay for the step-size.

**Proposition 5.8 (Convergence of SGD for strongly-convex problems)** *Assume that $F$ is convex, $B$-Lipschitz and that $F + \frac{\mu}{2}\|\cdot\|_2^2$ admits a (necessarily unique) minimizer $\theta_*$. Assume that the stochastic gradient $g$ satisfies (H-1) and (H-2). Then, choosing $\gamma_t = 1/(\mu t)$, the iterates $(\theta_t)_{t\geqslant 0}$ of SGD from Eq. (5.21) satisfy*

$$\mathbb{E}\big[G(\bar\theta_t) - G(\theta_*)\big] \leqslant \frac{2B^2(1+\log t)}{\mu t},$$

*where $\bar\theta_t = \frac{1}{t}\sum_{s=1}^t \theta_{s-1}$.*

**Proof** The beginning of the proof is essentially the same as for convex problems, leading to (with the new terms in blue):

$$\begin{aligned}\mathbb{E}\big[\|\theta_t-\theta_*\|_2^2\big] &= \mathbb{E}\big[\|\theta_{t-1}-\gamma_t(g_t(\theta_{t-1})+\mu\theta_{t-1})-\theta_*\|_2^2\big]\\ &= \mathbb{E}\big[\|\theta_{t-1}-\theta_*\|_2^2\big]-2\gamma_t\mathbb{E}\big[(g_t(\theta_{t-1})+\mu\theta_{t-1})^\top(\theta_{t-1}-\theta_*)\big]\\ &\qquad +\gamma_t^2\mathbb{E}\big[\|g_t(\theta_{t-1})+\mu\theta_{t-1}\|_2^2\big].\end{aligned}$$

From the iterations in Eq. (5.21), we see that $\theta_t = (1-\gamma_t\mu)\theta_{t-1} + \gamma_t\mu\big[-\frac{1}{\mu}g_t(\theta_{t-1})\big]$ is a convex combination of gradients divided by $-\mu$, and thus $\|g_t(\theta_{t-1}) + \mu\theta_{t-1}\|_2^2$ is always less than $4B^2$. Thus

$$\mathbb{E}\big[\|\theta_t-\theta_*\|_2^2\big] \leqslant \mathbb{E}\big[\|\theta_{t-1}-\theta_*\|_2^2\big] - 2\gamma_t\mathbb{E}\big[G'(\theta_{t-1})^\top(\theta_{t-1}-\theta_*)\big] + 4\gamma_t^2B^2.$$

Therefore, combining with the inequality coming from strong convexity $G(\theta_{t-1}) - G(\theta_*) + \frac{\mu}{2}\|\theta_{t-1} - \theta_*\|_2^2 \leqslant G'(\theta_{t-1})^\top(\theta_{t-1} - \theta_*)$ (see Eq. (5.9)), it follows

$$\gamma_t \mathbb{E}[G(\theta_{t-1}) - G(\theta_*)] \leqslant \frac{1}{2}\big((1-\gamma_t\mu)\mathbb{E}\|\theta_{t-1} - \theta_*\|^2 - \mathbb{E}\|\theta_t - \theta_*\|^2\big) + 2\gamma_t^2 B^2,$$

and thus, now using the specific step-size choice $\gamma_t = 1/(\mu t)$:

$$\begin{aligned}\mathbb{E}[G(\theta_{t-1}) - G(\theta_*)] &\leqslant \frac{1}{2}\big((\gamma_t^{-1} - \mu)\mathbb{E}\|\theta_{t-1} - \theta_*\|^2 - \gamma_t^{-1}\mathbb{E}\|\theta_t - \theta_*\|^2\big) + 2\gamma_t B^2, \\ &= \frac{1}{2}\big(\mu(t-1)\mathbb{E}\|\theta_{t-1} - \theta_*\|^2 - \mu t\mathbb{E}\|\theta_t - \theta_*\|^2\big) + \frac{2B^2}{\mu t}.\end{aligned}$$

Thus, we get a telescoping sum: summing between all indices between 1 and $t$, and using the bound $\sum_{s=1}^t \frac{1}{s} \leqslant 1 + \log t$, we get the desired result. ■

We can make the following observations:

- For smooth problems, we can show a similar bound of the form $O(\kappa/t)$. For quadratic problems, constant step-sizes can be used with averaging, leading to improved convergence rates (Bach and Moulines, 2013). See the exercise below.

  **Exercise 5.28** (♦) *We consider the minimization of* $F(\theta) = \frac{1}{2}\theta^\top H\theta - c^\top\theta$, *where* $H \in \mathbb{R}^{d\times}$ *is positive definite (and thus invertible). We consider the recursion* $\theta_t = \theta_{t-1} - \gamma[F'(\theta_{t-1}) + \varepsilon_t]$, *where all* $\varepsilon_t$*'s are independent, with zero mean and covariance matrix equal to* $C$. *Compute explicity* $\mathbb{E}\big[F(\theta_t) - F(\theta_*)\big]$, *and provide an upper-bound of* $\mathbb{E}\big[F(\bar\theta_t) - F(\theta_*)\big]$, *where* $\bar\theta_t = \frac{1}{t}\sum_{s=0}^{t-1}\theta_s$.

- The bound in $O(B^2/\mu t)$ is optimal for this class of problems. That is, as shown for example by Agarwal et al. (2009), among all algorithms that can query stochastic gradients, having a better convergence rate (up to some constants) is impossible (see Section 15.3).

- We note that for the same regularized problem, we could use a step size proportional to $DB/\sqrt{t}$ and obtain a bound proportional to $DB/\sqrt{t}$, which looks worse than $B^2/(\mu t)$ but can, in fact, be better when $\mu$ is very small.

  Note also the loss of adaptivity: the step-size now depends on the problem's difficulty (this was different for deterministic gradient descent). See experiments below for illustrations.

**Exercise 5.29** *With the same assumptions as Prop. 5.8, show that with the step-size* $\gamma_t = \frac{2}{\mu(t+1)}$, *and with* $\bar\theta_t = \frac{2}{t(t+1)}\sum_{s=1}^t s\theta_{s-1}$, *we have:* $\mathbb{E}\big[G(\bar\theta_t) - G(\theta_*)\big] \leqslant \frac{8B^2}{\mu(t+1)}$.

**Experiments.** We consider a simple binary classification problem with linear predictors in dimension $d = 40$ (inputs generated from a Gaussian distribution, with binary outputs obtained as the sign of a linear function with additive Gaussian noise), with $n = 400$ observations, and observed features with $\ell_2$-norm bounded by $R$. We consider the hinge

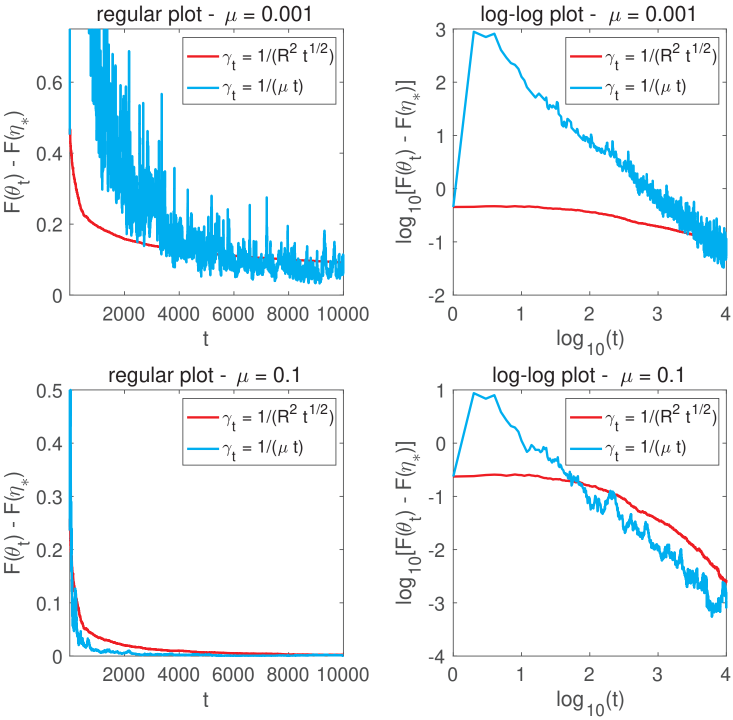


Figure 5.2: Comparison of step-sizes for SGD for the support vector machine, for two values of the regularization parameter $\mu$ (top: large $\mu$, bottom: small $\mu$). The performance is measured with a single run (hence the variability) on the excess training objective (left: regular plot, right: "log-log" plot).

loss with a squared $\ell_2$-regularizer $\frac{\mu}{2}\|\cdot\|_2^2$ (that is, the support vector machine presented in Section 4.1.2). We measure the excess training objective. We consider two values of $\mu$, and compare the two step-sizes $\gamma_t = 1/(R^2\sqrt{t})$ and $\gamma_t = 1/(\mu t)$ in Figure 5.2. We see that for large enough $\mu$, the strongly-convex step-size is better. This is not the case for small $\mu$.

The experiments above highlight the danger of a step-size equal to $1/(\mu t)$. In practice, it is often preferable to use $\gamma_t = \frac{1}{B^2\sqrt{t}+\mu t}$, as shown in the exercise below.

**Exercise 5.30** *(♦♦) With the same assumptions as in Prop. 5.8, with $\gamma_t = \frac{1}{B^2\sqrt{t}+\mu t}$, provide a convergence rate for the averaged iterate.*

### 5.4.2 Adaptive methods (♦)

The discussion on pre-conditioning for gradient descent on smooth functions at the end of Section 5.2.5 can be adapted to stochastic gradient methods for non-smooth problems. In this section, we highlight the potential gains and give references for precise results.

We focus on a linear prediction problem with i.i.d. features bounded in $\ell_2$-norm by $R$, and a convex $G$-Lipschitz-continuous loss function, in the setting of Prop. 5.7. For a constant step-size $\gamma$, in the proof of Prop. 5.7, we obtained an expected excess-risk equal to, starting from $\theta_0 = 0$,

$$\frac{1}{2\gamma t}\|\theta_*\|_2^2 + \frac{\gamma G^2}{2}\operatorname{tr}[\Sigma],$$

where $\Sigma = \mathbb{E}[\varphi(x)\varphi(x)^\top]$ is the covariance matrix of the features. Optimizing over $\gamma$ leads to the overall rate of $\frac{G\|\theta_*\|_2}{\sqrt{t}}\sqrt{\operatorname{tr}[\Sigma]}$.

Like done at the end of Section 5.2.5, pre-multiplying each gradient by the matrix $AA^\top$ is equivalent to minimizing the expectation of $\ell(y, \varphi(x)^\top A\tilde{\theta})$, which itself corresponds to replacing the feature map $\varphi$ by $A^\top\varphi$, and $\theta_*$ by $A^{-1}\theta_*$. The complexity bound above then becomes

$$\frac{1}{2\gamma t}\theta_*^\top(AA^\top)^{-1}\theta_* + \frac{\gamma G^2}{2}\operatorname{tr}[\Sigma AA^\top].$$

The matrix $M = (\gamma AA^\top)^{-1}$, which is the inverse of the matrix multiplying the gradient in the SGD iteration, can be optimized in the specific situation where we restrict the matrix $M$ to be diagonal with diagonal $m \in \mathbb{R}^d$. We then obtain the bound

$$\frac{1}{2t}\|\theta_*\|_\infty^2 \cdot \sum_{j=1}^d m_j + \frac{G^2}{2}\sum_{j=1}^d \frac{\Sigma_{jj}}{m_j},$$

with optimal $m_j$ equal to $\Sigma_{jj}^{1/2} G\sqrt{t}/\|\theta_*\|_\infty$ and an overall rate equal to $\frac{G\|\theta_*\|_\infty}{\sqrt{t}}\sum_{j=1}^d \Sigma_{jj}^{1/2}$, which can be substantially smaller than the corresponding rate with uniform $m$, proportional to $\frac{G\|\theta_*\|_\infty}{\sqrt{t}} d\sqrt{\sum_{j=1}^d \Sigma_{jj}}$; this in particular the case, where the $\Sigma_{jj}$'s have heterogeneous values.

In practice, we can estimate before running the learning algorithm the required elements of $\Sigma$, the (non-centered) covariance matrix of the features, and, more generally, the covariance of the gradients. These quantities can be estimated online, leading to the algorithms Adagrad (Duchi et al., 2011), or Adam (Kingma and Ba, 2014), which come with specific complexity bounds (see, e.g., Défossez et al., 2022).

### 5.4.3 Bias-variance trade-offs for least-squares (♦)

In this section, we consider the least-squares learning problems studied in Chapter 3, that is, we assume that we have i.i.d. observations $(x_i, y_i) \in \mathcal{X} \times \mathbb{R}$, for $i \geqslant 1$, assuming that there exists a feature map $\varphi : \mathcal{X} \to \mathbb{R}^d$ and $\theta_* \in \mathbb{R}^d$ such that $y_i = \varphi(x_i)^\top\theta_* + \varepsilon_i$, where $\varepsilon_i$ has mean zero and variance $\sigma^2$, and is independent of $x_i$. The goal of this section is to relate the performance of single-pass SGD to (regularized) empirical risk minimization studied in Section 3.3 and Section 3.6, and to study the impact of noise in SGD precisely.

The SGD recursion, often referred to as the least-mean-squares (LMS) recursion, can be written as, with a constant step-size:

$$\theta_t = \theta_{t-1} - \gamma(\theta_{t-1}^\top\varphi(x_t) - y_t)\varphi(x_t) = \theta_{t-1} - \gamma(\theta_{t-1}^\top\varphi(x_t) - \theta_*^\top\varphi(x_t) - \varepsilon_t)\varphi(x_t),$$

leading to

$$\theta_t - \theta_* = (I - \gamma\varphi(x_t)\varphi(x_t)^\top)(\theta_{t-1} - \theta_*) + \gamma\varepsilon_t\varphi(x_t). \tag{5.22}$$

Thus, like in the deterministic case in Section 5.2.1, we obtain a linear dynamical system, this time with random coefficients.

**Classical analysis.** We can first use a similar proof as in previous sections, that is, expanding Eq. (5.22),

$$\begin{aligned}\|\theta_t - \theta_*\|_2^2 &= \|\theta_{t-1} - \theta_*\|_2^2 + \|\gamma\varphi(x_t)\varphi(x_t)^\top(\theta_{t-1} - \theta_*)\|_2^2 \\ &\qquad -2\gamma(\theta_{t-1} - \theta_*)^\top\varphi(x_t)\varphi(x_t)^\top(\theta_{t-1} - \theta_*) + \|\gamma\varepsilon_t\varphi(x_t)\|_2^2 \\ &\qquad +2\gamma\varepsilon_t\varphi(x_t)^\top(I - \gamma\varphi(x_t)\varphi(x_t)^\top)(\theta_{t-1} - \theta_*),\end{aligned}$$

leading to, with $\mathcal{F}_{t-1}$ the information up to time $t-1$ (generated by $x_1, y_1, \dots, x_{t-1}, y_{t-1}$), and using that $\|\varphi(x_t)\|_2^2 \leqslant R^2$ almost surely, and the inequality $\Sigma = \mathbb{E}\big[\varphi(x_t)\varphi(x_t)^\top\big]$, and $\mathbb{E}\big[\|\varphi(x_t)\|_2^2\varphi(x_t)\varphi(x_t)^\top\big] \preccurlyeq \Sigma$:

$$\mathbb{E}\big[\|\theta_t - \theta_*\|_2^2\big|\mathcal{F}_{t-1}\big] \leqslant \|\theta_{t-1} - \theta_*\|_2^2 + (\gamma^2R^2 - 2\gamma)(\theta_{t-1} - \theta_*)^\top\Sigma(\theta_{t-1} - \theta_*) + \gamma^2\sigma^2R^2.$$

This leads to, with $F(\theta) - F(\theta_*) = \frac{1}{2}(\theta - \theta_*)^\top\Sigma(\theta - \theta_*)$, for $\gamma \leqslant 1/R^2$,

$$\mathbb{E}\big[F(\theta_{t-1}) - F(\theta_*)\big] \leqslant \frac{1}{2\gamma}\Big(\mathbb{E}\big[\|\theta_{t-1} - \theta_*\|_2^2\big] - \mathbb{E}\big[\|\theta_t - \theta_*\|_2^2\big]\Big) + \frac{\gamma\sigma^2R^2}{2},$$

and thus, for the average $\bar\theta_t = \frac{1}{t}\sum_{s=1}^t \theta_{s-1}$, using Jensen's inequality,

$$\mathbb{E}\big[F(\bar\theta_t)\big] - F(\theta_*)\big] \leqslant \frac{1}{2\gamma t}\|\theta_0 - \theta_*\|_2^2 + \frac{\gamma\sigma^2R^2}{2},$$

which is a similar result to the non-smooth case but with an explicit bias/variance decomposition where the noise variance $\sigma^2$ explicitly appears, as well as the norm of $\theta_*$. Note that it requires the step-size to depend on the number of total iterations to obtain convergence.

However, for least-squares, a finer analysis can be performed, allowing explicitly for constant step-sizes and a clear relationship with generalization bounds for least-squares outlined in Chapter 3.

**Finer analysis of the LMS recursion (♦♦).** A detailed analysis of the LMS recursion in Eq. (5.22) is out of the scope of this textbook. However, a simplified recursion with essentially the same behavior can be analyzed with simple linear algebra tools. To obtain this simplified recursion, we rewrite Eq. (5.22) as

$$\theta_t - \theta_* = (I - \gamma\Sigma)(\theta_{t-1} - \theta_*) + \gamma\varepsilon_t\varphi(x_t) + \gamma(\Sigma - \varphi(x_t)\varphi(x_t)^\top)(\theta_{t-1} - \theta_*),$$

which is the recursion of the expected risk, corresponding to the term $(I - \gamma\Sigma)(\theta_{t-1} - \theta_*)$, plus additional stochastic terms with zero conditional mean. One of them, $\gamma\varepsilon_t\varphi(x_t)$ is purely “additive” (i.e., it does not depend on $\theta_{t-1}$) and has a constant non-zero variance, while the other one, $\gamma(\Sigma - \varphi(x_t)\varphi(x_t)^\top)(\theta_{t-1} - \theta_*)$ is multiplicative and has a variance that will go to zero as iterates converge to $\theta_*$. The simplified recursion ignores that term, and we now study the recursion (started at $\eta_0 = \theta_0$):

$$\eta_t - \theta_* = (I - \gamma\Sigma)(\eta_{t-1} - \theta_*) + \gamma\varepsilon_t\varphi(x_t), \tag{5.23}$$

which also corresponds to replacing $\varphi(x_t)\varphi(x_t)^\top$ in Eq. (5.22) by its expectation $\Sigma$ .

We can then explicitly unroll the recursion as:

$$\eta_t - \theta_* \;=\; (I - \gamma\Sigma)^t(\eta_0 - \theta_*) + \sum_{u=1}^{t} \gamma\varepsilon_u (I - \gamma\Sigma)^{t-u}\varphi(x_u),$$

with two parts, one which only depends on the initialization, that is, $(I - \gamma\Sigma)^t(\eta_0 - \theta_*)$, which is precisely the deterministic recursion from Section 5.2.1, and we refer to it as the “bias” part, and a part that depends on the noise variables $\varepsilon_u$, $u = 1, \dots, t$, which we refer to as the “variance” part. Assuming these noise variables are independent from $x$, the two parts can be considered totally independently when taking expectations.

We then have, for the averaged iterates:

$$\begin{aligned}
\bar\eta_t^{(\text{bias})} - \theta_* &= \frac{1}{t}\sum_{v=0}^{t-1}(I - \gamma\Sigma)^v(\eta_0 - \theta_*) = \frac{1}{t}(\gamma\Sigma)^{-1}\big[I - (I - \gamma\Sigma)^t\big](\eta_0 - \theta_*) \\
\bar\eta_t^{(\text{var})} - \theta_* &= \frac{1}{t}\sum_{v=1}^{t-1}\sum_{u=1}^{v} \gamma\varepsilon_u (I - \gamma\Sigma)^{v-u}\varphi(x_u) = .\frac{\gamma}{t}\sum_{u=1}^{t-1}\sum_{v=u}^{t-1}(I - \gamma\Sigma)^{v-u}\varepsilon_u\varphi(x_u) \\
&= \frac{1}{t}\sum_{u=1}^{t-1}\Sigma^{-1}\big[I - (I - \gamma\Sigma)^{t-u}\big]\varepsilon_u\varphi(x_u),
\end{aligned}$$

leading to

$$\begin{aligned}
\big\|\bar\eta_t^{(\text{bias})} - \theta_*\big\|_\Sigma^2 &= \frac{1}{t^2}(\eta_0 - \theta_*)^\top(\gamma\Sigma)^{-2}\big[I - (I - \gamma\Sigma)^t\big]^2\Sigma(\eta_0 - \theta_*) \\
&\leqslant \frac{1}{\gamma^2 t^2}(\eta_0 - \theta_*)^\top\Sigma^{-1}(\eta_0 - \theta_*), \\
\mathbb{E}\Big[\big\|\bar\eta_t^{(\text{var})} - \theta_*\big\|_\Sigma^2\Big] &= \frac{\sigma^2}{\gamma t^2}\sum_{u=1}^{t-1}\operatorname{tr}\Big[\Sigma^2\Sigma^{-2}\big[I - (I - \gamma\Sigma)^{t-u}\big]^2\Big] \leqslant \frac{\sigma^2 d}{t}.
\end{aligned}$$

We thus obtain two terms, the variance term in $\frac{\sigma^2 d}{t}$, which is present because the optimal prediction is not equal to the response, and the bias term in $\frac{1}{\gamma^2 t^2}(\eta_0 - \theta_*)^\top\Sigma^{-1}(\eta_0 - \theta_*)$, which corresponds to the forgetting of initial conditions. It is worth comparing to the same quantities for the non-averaged iterates: the bias term is upper-bounded by (using the same constants) $\frac{1}{\gamma^2 t^2}(\eta_0 - \theta_*)^\top\Sigma^{-1}(\eta_0 - \theta_*)$, but it is typically faster when the lowest eigenvalue of $\Sigma$ is strictly positive. The variance term is only of order $\gamma\sigma^2\operatorname{tr}[\Sigma]$ for the variance term (thus. with no convergence). This is illustrated in Figure 5.3.

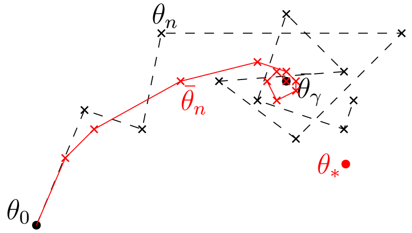


Figure 5.3: The iterates of SGD form a Markov chain, which is homogeneous when the step-size $\gamma$ is constant. It typically converges to a stationary distribution with expectation $\bar\theta_\gamma$, which happens to be the global minimum $\theta_*$ for quadratic costs (and with a deviation of $\gamma^2$ in general). The non-average iterates go from the initial condition $\theta_0$ to the vicinity of $\bar\theta_\gamma$, while the averaged iterates converge to that expectation $\bar\theta_\gamma$.

When $t = n$ iterations are performed, these should be compared to the excess risk for the least-squares estimators defined in Section 3.3 obtained by minimizing the empirical risk (only with the fixed design assumption). The variance term is the same as $\sigma^2 d/n = O(1/n)$, while the bias term is in $O(1/n^2)$ and seems smaller in the dependence in $n$. However, in high-dimensional problems, it can start to be larger for small $n$, highlighting the impact of forgetting initial conditions (see, e.g., Défossez and Bach, 2015).

The analysis provided in this section can be extended in several ways, for the "true" multiplicative noise, with similar results (Bach and Moulines, 2013; Défossez and Bach, 2015), to obtain dimension-free results akin to Section 3.6 (Dieuleveut and Bach, 2016; Dieuleveut et al., 2017), and to go beyond least-squares regression by studying logistic regression (Bach, 2014).

### 5.4.4 Variance reduction (♦)

We consider a finite sum $F(\theta) = \frac{1}{n}\sum_{i=1}^n f_i(\theta)$, where each $f_i$ is $R^2$-smooth (for example, logistic regression with features bounded by $R$ in $\ell_2$-norm), and which is such that $F$ is $\mu$-strongly convex (for example by adding $\frac{\mu}{2}\|\theta\|_2^2$ to each $f_i$, or by benefitting from the strong convexity of the sum). We denote by $\kappa = R^2/\mu$ the condition number of the problem (note that it is more significant than $L/\mu$, where $L$ is the smoothness constant of $F$).

Using SGD, the convergence rate has been shown to be $O(\kappa/t)$ in Section 5.4.1, with iterations of complexity $O(d)$, while for GD, the convergence rate is $O(\exp(-t/\kappa))$ (see Section 5.2.3), but each iteration has complexity $O(nd)$. We now present a result allowing exponential convergence with an iteration cost of $O(d)$.

The idea is to use a form of *variance reduction*, made possible by keeping in memory past gradients. We denote by $z_i^{(t)} \in \mathbb{R}^d$ the version of gradient $i$ stored at time $t$.

The SAGA algorithm (Defazio et al., 2014), which combines the earlier algorithms SAG (Schmidt et al., 2017) and SVRG (Johnson and Zhang, 2013; Zhang et al., 2013), works as follows: at every iteration, an index $i(t)$ is selected uniformly at random in

$\{1, \dots, n\}$, and we perform the iteration

$$\theta_t = \theta_{t-1} - \gamma \Big[ f'_{i(t)}(\theta_{t-1}) + \frac{1}{n} \sum_{i=1}^n z_i^{(t-1)} - z_{i(t)}^{(t-1)} \Big],$$

with $z_{i(t)}^{(t)} = f'_{i(t)}(\theta_{t-1})$ and all others $z_i^{(t)}$ left unchanged (i.e., the same as $z_i^{(t-1)}$). In words, we add to the update with the stochastic gradient $f'_{i(t)}(\theta_{t-1})$ the corrective term $\frac{1}{n} \sum_{i=1}^n z_i^{(t-1)} - z_{i(t)}^{(t-1)}$, which has zero expectation with respect to $i(t)$. Thus, since the expectation of $f'_{i(t)}(\theta_{t-1})$ with respect to $i(t)$ is equal to the full gradient $F'(\theta)$, the update is *unbiased* like for regular SGD. The goal is to reduce its variance.

The idea behind variance reduction is that if the random variable $z_{i(t)}^{(t-1)}$ (only considering the source of randomness coming from $i(t)$) is positively correlated with $f'_{i(t)}(\theta_{t-1})$, then the variance is reduced, and larger step-sizes can be used.

As the algorithm converges, then $z_i^{(t)}$ converges to $f'_i(\eta_*)$ (the individual gradient at optimum). We will show that *simultaneously* $\theta_t$ converges to $\eta_*$ *and* $z_i^{(t)}$ converges to $f'_i(\eta_*)$ for all $i$, all at the same speed.

**Proposition 5.9 (Convergence of SAGA)** *If initializing with $z_i^{(0)} = f'_i(\theta_0)$ at the initial point $\theta_0 \in \mathbb{R}^d$, for all $i \in \{1, \dots, n\}$, we have, for the choice of step-size $\gamma = \frac{1}{4R^2}$:*

$$\mathbb{E}\big[\|\theta_t - \eta_*\|_2^2\big] \leqslant \Big(1 - \min\Big\{\frac{1}{3n}, \frac{3\mu}{16R^2}\Big\}\Big)^t \big(1 + \frac{n}{4}\big) \|\theta_0 - \eta_*\|_2^2. \tag{5.24}$$

**Proof** (♦♦) The proof consists in finding a Lyapunov function that decays along iterations.

**Step 1.** We first try our "usual" Lyapunov function, making the differences $\|z_i^{(t)} - f'_i(\eta_*)\|_2^2$ appear, with the update $\theta_t = \theta_{t-1} - \gamma\omega_t$, with $\omega_t = \big[f'_{i(t)}(\theta_{t-1}) + \frac{1}{n}\sum_{i=1}^n z_i^{(t-1)} - z_{i(t)}^{(t-1)}\big]$,

$$\begin{aligned} \|\theta_t - \eta_*\|_2^2 &= \|\theta_{t-1} - \eta_*\|_2^2 - 2\gamma(\theta_{t-1} - \eta_*)^\top \omega_t + \gamma^2 \big\|\omega_t\big\|_2^2 \text{ by expanding the square,} \\ \mathbb{E}_{i(t)} \|\theta_t - \eta_*\|_2^2 &= \|\theta_{t-1} - \eta_*\|_2^2 - 2\gamma(\theta_{t-1} - \eta_*)^\top F'(\theta_{t-1}) \\ &\qquad + \gamma^2 \mathbb{E}_{i(t)} \Big\| f'_{i(t)}(\theta_{t-1}) + \frac{1}{n}\sum_{i=1}^n z_i^{(t-1)} - z_{i(t)}^{(t-1)} \Big\|_2^2, \end{aligned}$$

using the unbiasedness of the stochastic gradient. We further get

$$\begin{aligned} \mathbb{E}_{i(t)} \|\theta_t - \eta_*\|_2^2 &\leqslant \|\theta_{t-1} - \eta_*\|_2^2 - 2\gamma(\theta_{t-1} - \eta_*)^\top F'(\theta_{t-1}) \\ &\quad + 2\gamma^2 \mathbb{E}_{i(t)} \big\| f'_{i(t)}(\theta_{t-1}) - f'_{i(t)}(\eta_*) \big\|_2^2 + 2\gamma^2 \mathbb{E}_{i(t)} \big\| f'_{i(t)}(\eta_*) - z_{i(t)}^{(t-1)} + \frac{1}{n}\sum_{i=1}^n z_i^{(t-1)} \big\|_2^2, \end{aligned}$$

using $\|a+b\|_2^2 \leqslant 2\|a\|_2^2 + 2\|b\|_2^2$. To bound $\mathbb{E}_{i(t)}\big\|f'_{i(t)}(\theta_{t-1}) - f'_{i(t)}(\eta_*)\big\|_2^2$, we use co-coercivity of all functions $f_i$ (see Prop. 5.4), to get:

$$\begin{aligned}
\mathbb{E}_{i(t)}\big\|f'_{i(t)}(\theta_{t-1}) - f'_{i(t)}(\eta_*)\big\|_2^2 &= \frac{1}{n}\sum_{i=1}^n \big\|f'_i(\theta_{t-1}) - f'_i(\eta_*)\big\|_2^2 \\
&\leqslant \frac{1}{n}\sum_{i=1}^n R^2[f'_i(\theta_{t-1}) - f'_i(\eta_*)]^\top(\theta_{t-1} - \theta_*) \\
&\leqslant R^2 F'(\theta_{t-1})^\top(\theta_{t-1} - \eta_*) \text{ since } \sum_{i=1}^n f'_i(\eta_*) = 0. \quad (5.25)
\end{aligned}$$

To bound $\mathbb{E}_{i(t)}\big\|f'_{i(t)}(\eta_*) - z_{i(t)}^{(t-1)} + \frac{1}{n}\sum_{i=1}^n z_i^{(t-1)}\big\|_2^2$, we use $\mathbb{E}_{i(t)}\|Z - \mathbb{E}_{i(t)}Z\|_2^2 \leqslant \mathbb{E}_{i(t)}\|Z\|_2^2$. We thus get

$$\begin{aligned}
\mathbb{E}_{i(t)}\|\theta_t - \eta_*\|_2^2 &\leqslant \|\theta_{t-1} - \eta_*\|_2^2 - 2\gamma(\theta_{t-1} - \eta_*)^\top F'(\theta_{t-1}) + 2\gamma^2 R^2(\theta_{t-1} - \eta_*)^\top F'(\theta_{t-1}) \\
&\qquad + 2\gamma^2\frac{1}{n}\sum_{i=1}^n \big\|f'_i(\eta_*) - z_i^{(t-1)}\big\|_2^2, \\
&\leqslant \|\theta_{t-1} - \eta_*\|_2^2 - 2\gamma(1 - \gamma R^2)(\theta_{t-1} - \eta_*)^\top F'(\theta_{t-1}) \\
&\qquad + 2\frac{\gamma^2}{n}\sum_{i=1}^n \big\|f'_i(\eta_*) - z_i^{(t-1)}\big\|_2^2.
\end{aligned}$$

**Step 2.** We see the term $\sum_{i=1}^n \big\|f'_i(\eta_*) - z_i^{(t-1)}\big\|_2^2$ appearing, so we try to study how it varies across iterations. We have, by definition of the updates of the vectors $z_i^{(t)}$:

$$\begin{aligned}
\sum_{i=1}^n \big\|f'_i(\eta_*) - z_i^{(t)}\big\|_2^2 &= \sum_{i=1}^n \big\|f'_i(\eta_*) - z_i^{(t-1)}\big\|_2^2 \\
&\qquad - \big\|f'_{i(t)}(\eta_*) - z_{i(t)}^{(t-1)}\big\|_2^2 + \big\|f'_{i(t)}(\eta_*) - f'_{i(t)}(\theta_{t-1})\big\|_2^2.
\end{aligned}$$

Taking expectations with respect to $i(t)$, we get

$$\begin{aligned}
\mathbb{E}_{i(t)}\Big[\sum_{i=1}^n \big\|f'_i(\eta_*) - z_i^{(t)}\big\|_2^2\Big] &= \Big(1 - \frac{1}{n}\Big)\sum_{i=1}^n \big\|f'_i(\eta_*) - z_i^{(t-1)}\big\|_2^2 + \frac{1}{n}\sum_{i=1}^n \big\|f'_i(\eta_*) - f'_i(\theta_{t-1})\big\|_2^2 \\
&\leqslant \Big(1 - \frac{1}{n}\Big)\sum_{i=1}^n \big\|f'_i(\eta_*) - z_i^{(t-1)}\big\|_2^2 + R^2(\theta_{t-1} - \eta_*)^\top F'(\theta_{t-1}),
\end{aligned}$$

where we use the bound in Eq. (5.25). Thus, for a positive number $\Delta$ to be chosen later,

$$\begin{aligned}
&\mathbb{E}_{i(t)}\Big[\|\theta_t - \eta_*\|_2^2 + \Delta\sum_{i=1}^n \big\|f'_i(\eta_*) - z_i^{(t)}\big\|_2^2\Big] \\
&\leqslant \|\theta_{t-1} - \eta_*\|_2^2 - 2\gamma\Big(1 - \gamma R^2 - \frac{R^2\Delta}{2\gamma}\Big)(\theta_{t-1} - \eta_*)^\top F'(\theta_{t-1}) \\
&\qquad + \Big[2\frac{\gamma^2}{n\Delta} + (1 - 1/n)\Big]\Delta\sum_{i=1}^n \big\|f'_i(\eta_*) - z_i^{(t-1)}\big\|_2^2.
\end{aligned}$$

With $\Delta = 3\gamma^2$ and $\gamma = \frac{1}{4R^2}$, we get $1 - \gamma R^2 - \frac{R^2\Delta}{2\gamma} = \frac{3}{8}$ and $2\frac{\gamma^2}{n\Delta} = \frac{2}{3n}$. Moreover, using the identity $(\theta_{t-1} - \eta_*)^\top F'(\theta_{t-1}) \geqslant \mu\|\theta_{t-1} - \eta_*\|_2^2$ as a consequence of strong convexity, we then get:

$$\begin{aligned}\mathbb{E}_{i(t)}\Big[\|\theta_t - \eta_*\|_2^2 + \Delta\sum_{i=1}^n \big\|f_i'(\eta_*) - z_i^{(t)}\big\|_2^2\Big] \quad &\leqslant \quad \Big(1 - \min\Big\{\frac{1}{3n}, \frac{3\mu}{16R^2}\Big\}\Big)\Big[\|\theta_{t-1} - \eta_*\|_2^2 \\ &\qquad + \Delta\sum_{i=1}^n \big\|f_i'(\eta_*) - z_i^{(t-1)}\big\|_2^2\Big].\end{aligned}$$

Thus

$$\mathbb{E}\big[\|\theta_t - \eta_*\|_2^2\big] \leqslant \Big(1 - \min\Big\{\frac{1}{3n}, \frac{3\mu}{16R^2}\Big\}\Big)^t \Big[\|\theta_0 - \eta_*\|_2^2 + \frac{3}{16R^4}\sum_{i=1}^n \big\|f_i'(\eta_*) - z_i^{(0)}\big\|_2^2\Big].$$

If initializing with $z_i^{(0)} = f_i'(\theta_0)$, we get the desired bound by using the Lipschitz-continuity of each $f_i'$, which leads to $(1 + \frac{3n}{16})\|\theta_0 - \eta_*\|_2^2 \leqslant (1 + \frac{n}{4})\|\theta_0 - \eta_*\|_2^2$. This leads to the final bound in Eq. (5.24). ■

We can make the following observations:

- The contraction rate after one iteration is $\big(1 - \min\big\{\frac{1}{3n}, \frac{3\mu}{16R^2}\big\}\big)$, which is less than $\exp\big(-\min\big\{\frac{1}{3n}, \frac{3\mu}{16R^2}\big\}\big)$. Thus, after an "effective pass" over the data, that is, $n$ iterations, the contracting rate is $\exp\big(-\min\big\{\frac{1}{3}, \frac{3\mu n}{16R^2}\big\}\big)$. It is only an effective pass because after we sample $n$ indices with replacement, we will not see all functions (while some will be seen several times).

  In order to have a contracting effect of $\varepsilon$, that is, having $\|\theta_t - \eta_*\|_2^2 \leqslant \varepsilon\|\theta_0 - \eta_*\|_2^2$, we need to have $\exp\big(-t\min\big\{\frac{1}{3n}, \frac{3\mu}{16R^2}\big\}\big)2n \leqslant \varepsilon$, which is equivalent to $t \geqslant \max\big\{3n, \frac{16R^2}{3\mu}\big\}\log\frac{2n}{\varepsilon}$. It just suffices to have $t \geqslant \big(3n + \frac{16R^2}{3\mu}\big)\log\frac{2n}{\varepsilon}$, and thus the running time complexity is equal to $d$ times the minimal number, that is

  $$d\Big(3n + \frac{16R^2}{3\mu}\Big)\log\frac{2n}{\varepsilon}.$$

  This is to be contrasted with batch gradient descent with step-size $\gamma = 1/R^2$ (which is the simplest step-size that can be computed easily), whose complexity is

  $$dn\frac{R^2}{\mu}\log\frac{1}{\varepsilon}.$$

  We replace the product of $n$ and condition number $\kappa = \frac{R^2}{\mu}$ by a sum, which is significant where $\kappa$ is large.

- Multiple extensions of this result are available, such as a rate for non-strongly-convex functions, adaptivity to strong-convexity, proximal extensions, and acceleration. It is also worth mentioning that the need to store past gradients can be alleviated (see Gower et al., 2020, for more details).

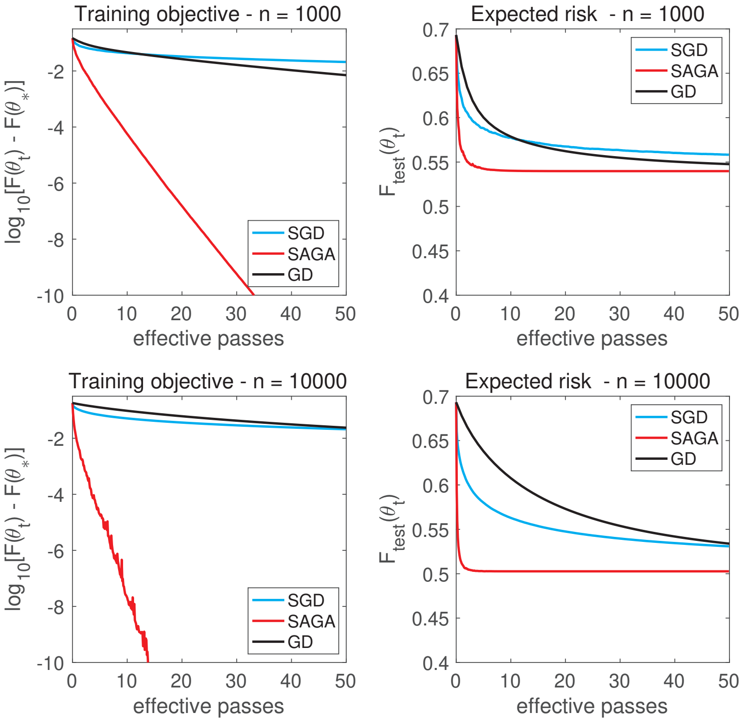


Figure 5.4: Comparison of stochastic gradient algorithms for logistic regression. Top: $n = 1000$, bottom: $n = 10000$. Left: training objective in semi-log plot, right: expected risk estimated with $n$ test points.

- Note that these fast algorithms allow very small optimization errors and that the best testing risks will typically be obtained after a few (10 to 100) passes.

**Experiments.** We consider $\ell_2$-regularized logistic regression, and we compare GD, SGD, and SAGA, all with their corresponding step-sizes coming from the theoretical analysis, with two values of $n$. We use a simple binary classification problem with linear predictors in dimension $d = 40$ (inputs generated from a Gaussian distribution, with binary outputs obtained as the sign of a linear function with additive Gaussian noise), with two different numbers of observations $n$, and regularization parameter $\mu = R^2/n$. See Figure 5.4 (top: small $n$, left: large $n$). We see that for early iterations, SGD dominates GD, while for larger numbers of iterations, GD is faster. This last effect is not seen for large numbers of observations (right), where SGD always dominates GD. SAGA gets to machine precision after 50 effective passes over the data in the two cases. Note also the better performance on the testing data.

## 5.5 Conclusion

**Convex finite-dimensional problems.** We can now provide a summary of convergence rates below, with the main rates we have seen in this chapter (and some that we have not seen) for convex objective functions. We separate between convex and strongly convex, and between smooth and non-smooth, as well as between deterministic and stochastic methods. Below, $L$ is the smoothness constant, $\mu$ the strong convexity constant, $B$ the Lipschitz constant, and $D$ the distance to optimum at initialization.

| | convex | strongly convex |
|---|---|---|
| nonsmooth | deterministic: $BD/\sqrt{t}$<br>stochastic: $BD/\sqrt{t}$ | deterministic: $B^2/(t\mu)$<br>stochastic: $B^2/(t\mu)$ |
| smooth | deterministic: $LD^2/t^2$<br>stochastic: $LD^2/\sqrt{t}$<br>finite sum: $n/t$ | deterministic: $\exp(-t\sqrt{\mu/L})$<br>stochastic: $L/(t\mu)$<br>finite sum: $\exp(-\min\{1/n, \mu/L\}t)$ |

The convergence rates are often written as a number $t$ of accesses to individual gradients to achieve excess function values of $\varepsilon$. This corresponds to inverting each formula for $\varepsilon$ as a function of $t$ to a formula for $t$ as a function of $\varepsilon$. This leads to the following table:

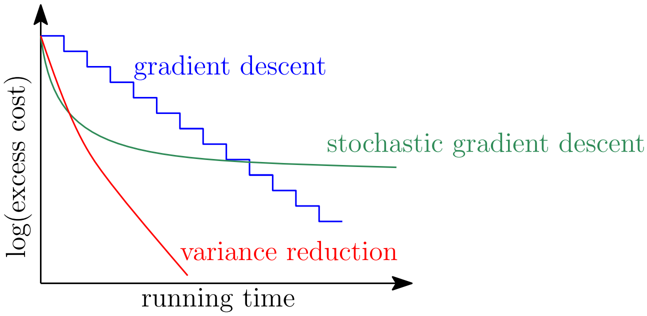
| | convex | strongly convex |
|---|---|---|
| nonsmooth | deterministic: $(BD)^2/\varepsilon^2$<br>stochastic: $(BD)^2/\varepsilon^2$ | deterministic: $B^2/(\varepsilon\mu)$<br>stochastic: $B^2/(\varepsilon\mu)$ |
| smooth | deterministic: $\sqrt{L}D/\sqrt{\varepsilon}$<br>stochastic: $(LD^2)^2/\varepsilon^2$<br>finite sum: $n/\varepsilon$ | deterministic: $\exp(-t\sqrt{\mu/L})$<br>stochastic: $L/(\varepsilon\mu)$<br>finite sum: $\max\{n, L/\mu\}\log(1/\varepsilon)$ |

⚠ Like in the rest of the book, where we obtain explicit convergence rates, the homogeneity of all quantities can be checked (see exercise below). In the context optimization, this ensures that algorithms are invariant by change of variable $\theta \to \alpha\theta$ for $\alpha \neq 0$.

**Exercise 5.31** *Check the homogeneity of all quantities of this section (step-size and convergence rates).*

Note that many important themes in optimization have been ignored, such as Frank-Wolfe methods (presented in Chapter 9), coordinate descent, or duality. See Nesterov (2018); Bubeck (2015) for further details. See also Chapter 7 and Chapter 9 for optimization methods for kernel methods and neural networks.

For strongly-convex smooth problems, the following toy figure also provides a good summary, with gradient descent being along a line in a semi-log plot (that is, exponential convergence) but with a staircase effect due to the lack of progress while computing the full gradient, SGD starting fast but having trouble reaching low optimization error, with variance reduction getting the best of both worlds, together with a faster rate of convergence than regular GD.



**Beyond finite-dimensional problems.** Supervised machine learning problems leading to finite-dimensional convex objective functions are essentially problems with prediction functions that are linear in their parameters, with a feature map that can be explicitly computed. In Chapter 7, we extend some of the algorithms seen in this chapter to features that are only available through dot-products $\varphi(x)^\top \varphi(x')$.

**Beyond convex problems.** Complexity bounds can be obtained beyond convex problems, as shown briefly in Section 5.2.6. However, they only certify that the gradient norm will go to zero, not that a global optimum has been approximately reached. Objective functions obtained from neural network training provide an important class of non-convex objective functions that we consider in Chapter 9.

**Generalization bounds: Rademacher or SGD?** In the last chapter, we have shown how to obtain generalization bounds for the constrained or regularized empirical risk minimizer. They relied on Rademacher complexities, which apply to all Lipschitz-continuous loss functions (not necessarily convex). However, they leave open how to obtain algorithmically such minimizers. In this section, we have not only seen algorithms to obtain such minimizers through gradient-based techniques but also single-pass SGD that directly provides the same generalization bound on unseen data for an efficient algorithm. We will see in Section 11.1.3 how this extends to the mirror descent framework to account for non-Euclidean geometries.

These two ways of obtaining generalization bounds will also be compared for multi-category classification in Chapter 13, where SGD-based bounds will lead to better bounds.