Chapter XII: First Order Methods for Large Scale Problems

In the previous chapter, we have seen that many problems that arise in the fields of machine learning and signal processing can be formulated as unconstrained optimization problems of the form

$$\underset{\boldsymbol{x}\in\mathbb{R}^n}{\textbf{minimize}}\ F(\boldsymbol{x}), \tag{1}$$

where F is a *non-smooth* convex function. For example, this is the case of the lasso problem

$$\underset{\boldsymbol{\theta}\in\mathbb{R}^n}{\textbf{minimize}}\ \|X\boldsymbol{\theta}-\boldsymbol{y}\|^2+\lambda\|\boldsymbol{\theta}\|_1,$$

or the soft-margin SVM:

$$\underset{\boldsymbol{w}\in\mathbb{R}^n,b\in\mathbb{R}}{\textbf{minimize}}\quad \sum_{i=1}^m \max(0,1-y_i(\boldsymbol{w}^T\boldsymbol{x}_i-b))+\lambda\|\boldsymbol{w}\|^2.$$

In the same vein, we could cite the low-rank matrix completion problem. Here, the input is an *incomplete matrix* $X\in(\mathbb{R}\cup\{\star\})^{m\times n}$, with a subset of defined entries $\Omega=\{(i,j):X_{ij}\neq\star\}$, and the goal is to find a matrix $Y\in\mathbb{R}^{m\times n}$ of lowest possible rank, such that the Y_{ij}'s approximate the X_{ij}'s for all $(i,j)\in\Omega$. A convex relaxation of this problem (leading to accurate data reconstruction, cf. [1]) is as follows:

$$\underset{Y\in\mathbb{R}^{m\times n}}{\textbf{minimize}}\quad \sum_{(i,j)\in\Omega}(X_{ij}-Y_{ij})^2+\lambda\|Y\|_*,$$

where $\|Y\|_*$ is the *nuclear norm* of Y, a non-smooth convex function that gives the sum of the singular values of Y.

While interior point methods are very efficient (and the method of choice) to solve such problems with up to a few tens of thousands of variables, Newton iterations become too costly for very large problems (with millions of variables). For problems involving *big-data*, we need algorithms that make use of first-order information only, in order to have *cheap iterations*. This comes at the price of a much slower convergence, but fortunately, near-optimal solutions can often be obtained after a reasonable amount of time. Anyway, when large datasets are involved, the data is often faulty, so solving a model to (true) optimality does not really make sense.

1 Gradient & Subgradient Methods

The most basic (and intuitive) idea to solve a problem of the form (1) is to use the *gradient descent*, which dates back to Cauchy (mid of 19th century): The idea is to start from $\boldsymbol{x}^{(0)}\in\mathbb{R}^n$, and at each iteration, we move in the direction of the gradient:

$$\boldsymbol{x}^{(k)}=\boldsymbol{x}^{(k-1)}-t_k\nabla F(\boldsymbol{x}^{(k-1)}),$$

where t_k is a suitable step size, which can be fixed over the iterations, or computed by backtracking line search. We defer the analysis of this algorithm to a later section, as it will be the special case of another, more general algorithm, which can also handle non-smooth functions F.

1.1 Subgradients

The gradient method raises the problem of non-smoothness: how can we even define a first-order method if the gradient of the function to minimize is not defined everywhere? For this, we must appeal to the concept of subgradients, which generalizes the concept of gradients to non-smooth functions:

> **Definition 1.** (Subgradient). We say that a vector $\boldsymbol{g}$ is a *subgradient* of $f : \mathbb{R}^n \to \mathbb{R}$ at $\boldsymbol{x} \in \mathbf{dom}\, f$, if
>
> $$f(\boldsymbol{z}) \geq f(\boldsymbol{x}) + \boldsymbol{g}^T(\boldsymbol{z} - \boldsymbol{x}), \quad \forall \boldsymbol{z} \in \mathbf{dom}\, f.$$
>
> Geometrically, this means that the vector $[\boldsymbol{g}, -1]^T$ defines a supporting hyperplane to $\mathbf{epi}\, f$ at $(\boldsymbol{x}, f(\boldsymbol{x}))$.

A function f is called subdifferentiable at $\boldsymbol{x}$ if its subdifferential

$$\partial f(\boldsymbol{x}) := \{\boldsymbol{g} \in \mathbb{R}^n : \boldsymbol{g} \text{ is a subgradient of } f \text{ at } \boldsymbol{x}\} = \bigcap_{\boldsymbol{z} \in \mathbf{dom}\, f} \{\boldsymbol{g} : f(\boldsymbol{z}) \geq f(\boldsymbol{x}) + \boldsymbol{g}^T(\boldsymbol{z} - \boldsymbol{x})\}$$

is nonempty. A basic property is that when a convex function f is differentiable at $\boldsymbol{x}$, then it is also subdifferentiable at $\boldsymbol{x}$, and its subdifferential is a singleton: $\partial f(\boldsymbol{x}) = \{\nabla f(\boldsymbol{x})\}$. From the definition, we also see that ∂f is always a convex closed set, because it is the intesection of (infinitely many) halfspaces.

Example:
Let $f : \mathbb{R} \to \mathbb{R}$, $f(x) = |x|$. Then, the subdifferential of f is given by

$$\partial f(x) = \begin{cases} \{-1\} & \text{if } x < 0; \\ [-1, 1] & \text{if } x = 0; \\ \{1\} & \text{if } x > 0. \end{cases}$$

$\boxed{\#1}$

Another important property, which is a simple consequence from the supporting hyperplane theorem, is that if the function f is convex, then it is subdifferentiable at all $\boldsymbol{x} \in \mathbf{int}\,\mathbf{dom}\, f$.

> **Theorem 1.** *Let f be convex. Then, $\boldsymbol{x}^*$ minimizes f over $\mathbb{R}^n$ if and only if $\mathbf{0} \in \partial f(\boldsymbol{x}^*)$.*

Proof. This is straightforward from the definition of a subgradient: $\mathbf{0} \in \partial f(\boldsymbol{x}^*) \iff f(\boldsymbol{z}) \geq f(\boldsymbol{x}^*)$, $\forall \boldsymbol{z} \in \mathbf{dom}\, f$. □

We conclude this section on subgradient with basic calculus rules for subdifferentials:

Proposition 2. *Let f,$f_1, \ldots, f_m$ be convex functions. Then,*

(i) [Nonnegative scaling]: $\partial(\alpha f)(\boldsymbol{x}) = \alpha \partial f(\boldsymbol{x})$, for all $\alpha \geq 0$.

(ii) [Sum]: $\partial(f_1 + \ldots + f_m)(\boldsymbol{x}) = \partial f_1(\boldsymbol{x}) + \ldots + \partial f_m(\boldsymbol{x})$.
(Note that the sum of the right hand side is a Minkowski sum of convex sets.

(iii) [Affine transformation]: $\partial\big(\boldsymbol{z} \mapsto f(A\boldsymbol{z} + b)\big)(\boldsymbol{x}) = A^T \partial f(A\boldsymbol{x} + b)$.

(iv) [Pointwise maximum]: Denote by g the pointwise maximum of the f_i's, i.e.,

$$g(\boldsymbol{x}) = \max_{i=1,\ldots,m} f_i(\boldsymbol{x}).$$

Then, $\partial g(\boldsymbol{x}) = \mathbf{conv}\left(\bigcup_{j \in A(\boldsymbol{x})} \partial f_j(\boldsymbol{x})\right)$, where $A(\boldsymbol{x})$ is the set of active functions *at $\boldsymbol{x}$, i.e., $A(\boldsymbol{x}) := \{j \in [m] : f_j(\boldsymbol{x}) = g(\boldsymbol{x})\}$. This property can be extended to pointwise supremums of infinitely many functions, provided mild additional technical condition hold.*

1.2 The subgradient method

The subgradient method is a direct adaptation of the gradient method, where the gradient $\nabla F(\boldsymbol{x})$ is replaced by any subgradient $\boldsymbol{g} \in \partial F(\boldsymbol{x})$:

$$\boldsymbol{x}^{(k)} = \boldsymbol{x}^{(k-1)} - \alpha_k \boldsymbol{g}^{(k-1)}, \qquad \text{for some } \boldsymbol{g}^{(k-1)} \in \partial F(\boldsymbol{x}^{(k-1)}).$$

We give several properties on this method, without proving it:

- Contrarily to the gradient method, the subgradient method is not a *descent* method, i.e., it is possible that $F(\boldsymbol{x}^{(k)}) > F(\boldsymbol{x}^{(k-1)})$, even for arbitrarily small step sizes.
- If exact or backtracking line search is used, the method can converge to a suboptimal point.
- However, convergence can be proved for several *offline rules* that select the step sizes α_k, e.g. constant step sizes ($\alpha_k = \alpha > 0, \forall k \in \mathbb{N}$) or nonsummable diminishing ($\alpha_k \geq 0$, $\lim\limits_{k\to\infty} \alpha_k = 0$, $\sum_{k=1}^{\infty} \alpha_k = \infty$, such as e.g. $\alpha_k = 1/k$).
- The convergence is slow: after k iteration, the best iterate seen so far typically satisfies

$$f(\boldsymbol{x}_{\text{best}}^{(k)}) \leq f(\boldsymbol{x}^*) + O(\frac{1}{\sqrt{k}}),$$

  which means that we need $O(\epsilon^2)$ iterations to find an $\epsilon-$suboptimal solution.

In the next section, we will see how to generalize the gradient method so it can handle non-smooth functions, while keeping its $O(1/k)$-convergence property.

2 The Proximal map

The proximal map of a convex function was defined by Moreau in 1965, and plays an important role in first-order methods for nonsmooth optimization.

> **Definition 2.** (Prox operator). Let $g : \mathbb{R}^n \to \mathbb{R} \cup \{\infty\}$ be a proper closed convex function (recall that g is *closed* if $\mathbf{epi}\, g$ is closed; *proper* means that $\mathbf{dom}\, g \neq \emptyset$). We define the proximal mapping of g by
>
> $$\mathbf{prox}_{\,g}(\boldsymbol{x}) := \operatorname*{argmin}_{\boldsymbol{u} \in \mathbb{R}^n} \quad g(\boldsymbol{u}) + \frac{1}{2}\|\boldsymbol{x} - \boldsymbol{u}\|^2.$$

In particular, we should note that the proximal operator generalizes the notion of projection over a convex set. Indeed, if I_C is the convex indicator function of a closed convex set C (i.e., $I_C(\boldsymbol{x}) = 0$ if $\boldsymbol{x} \in C$, and $I_C(\boldsymbol{x}) = \infty$ otherwise), then it is easy to see that $\mathbf{prox}_{\,I_C}(\boldsymbol{x})$ coincides with the projection of $\boldsymbol{x}$ onto C:

$$\mathbf{prox}_{\,I_C}(\boldsymbol{x}) = P_C(\boldsymbol{x}) = \operatorname*{argmin}_{\boldsymbol{u} \in C} \ \|\boldsymbol{u} - \boldsymbol{x}\|.$$

This definition requires some justification. Indeed, we must show that the minimizer of $h : \boldsymbol{u} \mapsto g(\boldsymbol{u}) + \frac{1}{2}\|\boldsymbol{x} - \boldsymbol{u}\|^2$ exists and is unique. Recall that a function f is $\nu-$strongly convex for some $\nu > 0$ if $f(\cdot) - \frac{\nu}{2}\|\cdot\|^2$ is convex. To prove that the prox operator is well defined, we are going to use the fact that h is is strongly convex with parameter $\nu = 1$, which is true because $h(\boldsymbol{u}) - \frac{1}{2}\|\boldsymbol{u}\|^2 = g(\boldsymbol{u}) - \boldsymbol{x}^T\boldsymbol{u} + \text{constant}$.

> **Lemma 3.** *Let f be a $\nu-$strongly convex function, and let $\boldsymbol{g} \in \partial f(\boldsymbol{x}_0)$. Then, for all $\boldsymbol{x} \in \mathbf{dom}\, f$,*
>
> $$f(\boldsymbol{x}) \geq f(\boldsymbol{x}_0) + \langle \boldsymbol{g}, \boldsymbol{x} - \boldsymbol{x}_0 \rangle + \frac{\nu}{2}\|\boldsymbol{x} - \boldsymbol{x}_0\|^2.$$

Proof. Define the function $F := f - \nu/2\|\cdot\|^2$. This function is convex (by ν-strong convexity of f), so $f = F + \nu/2\|\cdot\|^2$ is the sum of two convex functions, and we can use the rule for the subdifferentials: $\partial f(\boldsymbol{x}_0) = \partial F(\boldsymbol{x}_0) + \nu \boldsymbol{x}_0$. This shows that the vector $\boldsymbol{g} - \nu\boldsymbol{x}_0$ is a subgradient of F at $\boldsymbol{x}_0$, so for all $\boldsymbol{x} \in \mathbf{dom}\, F = \mathbf{dom}\, f$, it holds

$$\begin{aligned}
&F(\boldsymbol{x}) \geq F(\boldsymbol{x}_0) + \langle \boldsymbol{g} - \nu\boldsymbol{x}_0, \boldsymbol{x} - \boldsymbol{x}_0\rangle \\
\iff\ & f(\boldsymbol{x}) - \nu/2\|\boldsymbol{x}\|^2 \geq f(\boldsymbol{x}_0) - \nu/2\|\boldsymbol{x}_0\|^2 + (\boldsymbol{g} - \nu\boldsymbol{x}_0)^T(\boldsymbol{x} - \boldsymbol{x}_0) \\
\iff\ & f(\boldsymbol{x}) \geq f(\boldsymbol{x}_0) + \boldsymbol{g}^T(\boldsymbol{x} - \boldsymbol{x}_0) + \frac{\nu}{2}\|\boldsymbol{x} - \boldsymbol{x}_0\|^2.
\end{aligned}$$

□

Remark: The converse statement is also valid: we can prove that a function is ν-strongly convex if and only if the inequality of the lemma holds for all $\boldsymbol{x}_0 \in \mathbf{dom}\, \partial f$ and for all $\boldsymbol{g} \in \partial f(\boldsymbol{x}_0)$.

Then, we will need the following theorem to guarantee that the prox-opertor is well defined:

Theorem 4. *Let f be a proper closed, $\nu-$strongly convex function. Then f has a unique minimizer $\boldsymbol{x}^*$, and*

$$f(\boldsymbol{x}) \geq f(\boldsymbol{x}^*) + \frac{\nu}{2}\|\boldsymbol{x} - \boldsymbol{x}^*\|^2, \quad \forall \boldsymbol{x} \in \mathbf{dom}\, f.$$

Proof. For the existence of a minimizer, take a subgradient $\boldsymbol{g} \in \partial f(\boldsymbol{x}_0)$, where $\boldsymbol{x}_0 \in \mathbf{dom}\, f$. From the previous lemma, we know that $f(\boldsymbol{x}) \geq f(\boldsymbol{x}_0) + \boldsymbol{g}^T(\boldsymbol{x} - \boldsymbol{x}_0) + \frac{\nu}{2}\|\boldsymbol{x} - \boldsymbol{x}_0\|^2, \forall \boldsymbol{x} \in \mathbf{dom}\, f$. So, we can restrict our search for a minimizer within the sublevel set at level $f(\boldsymbol{x}_0)$ of this underestimator, which is a compact set. Finally, it is known that every closed convex function is lower-semicontinuous, and every lower-semicontinuous function has a minimizer over a compact set.

Now, we use the same subgradient inequality as above, but evaluated at a minimizer $\boldsymbol{x}^*$, where we can take the subgradient $\boldsymbol{g} = \boldsymbol{0} \in \partial f(\boldsymbol{x}^*)$, so $f(\boldsymbol{x}) \geq f(\boldsymbol{x}^*) + \boldsymbol{0}^T(\boldsymbol{x} - \boldsymbol{x}^*) + \frac{\nu}{2}\|\boldsymbol{x} - \boldsymbol{x}^*\|^2 = f(\boldsymbol{x}^*) + \frac{\nu}{2}\|\boldsymbol{x} - \boldsymbol{x}^*\|^2$ holds for all $\boldsymbol{x} \in \mathbf{dom}\, f$. This inequality can be used to show the uniqueness of the minimizer: If $\tilde{\boldsymbol{x}}$ is also a minimizer of f, then, $f(\tilde{\boldsymbol{x}}) = f(\boldsymbol{x}^*) \geq f(\boldsymbol{x}^*) + \frac{\nu}{2}\|\tilde{\boldsymbol{x}} - \boldsymbol{x}^*\|^2 \implies \|\tilde{\boldsymbol{x}} - \boldsymbol{x}^*\|^2 \leq 0 \implies \tilde{\boldsymbol{x}} = \boldsymbol{x}^*$. □

Theorem 5. *Let $g : \mathbb{R}^n \to \mathbb{R} \cup \{\infty\}$ be a proper closed convex function. Then,*

(i) $\mathbf{prox}_g(\boldsymbol{x}) \in \mathbb{R}^n$ *is well defined (i.e., it is a singleton) for all* $\boldsymbol{x} \in \mathbf{dom}\, g$.

(ii) $\boldsymbol{u} = \mathbf{prox}_g(\boldsymbol{x}) \iff \boldsymbol{x} - \boldsymbol{u} \in \partial g(\boldsymbol{u}) \iff \exists \boldsymbol{g}_{\boldsymbol{u}} \in \partial g(\boldsymbol{u}) : \boldsymbol{u} + \boldsymbol{g}_{\boldsymbol{u}} = \boldsymbol{x}$

(iii) $\boldsymbol{x}^*$ *is a minimizer of g iff* $\boldsymbol{x}^* = \mathbf{prox}_g(\boldsymbol{x}^*)$.

Proof. (i) follows from the previous theorem and the strong convexity of $h : \boldsymbol{u} \mapsto g(\boldsymbol{u}) + \frac{1}{2}\|\boldsymbol{x} - \boldsymbol{u}\|^2$. For (ii), we have that $\boldsymbol{u} = \mathbf{prox}_g(\boldsymbol{x})$ iff $\boldsymbol{0} \in \partial h(\boldsymbol{u}) = \partial g(\boldsymbol{u}) + \boldsymbol{u} - \boldsymbol{x} \iff \boldsymbol{x} - \boldsymbol{u} \in \partial g(\boldsymbol{u})$. From this, we get (iii) from $\boldsymbol{x}^* = \mathbf{prox}_g(\boldsymbol{x}^*) \iff \boldsymbol{x}^* - \boldsymbol{x}^* = \boldsymbol{0} \in \partial g(\boldsymbol{x}^*)$, which occurs iff $\boldsymbol{x}^*$ is a minimizer of g. □

The last statement of shows that $\boldsymbol{x}^*$ satisfies a *fixed-point* property, which can be useful for the analysis of some algorithms.

In general computing the proximal operator can be a quite difficult task (it can be as hard as solving the problem of minimizing g itself). But the prox-operator is available in closed-form for many interesting cases; for an catalog, cf. the website [2]. This includes the case of indicator functions for the following convex sets: boxes, hyperplanes, the unit simplex and the probability simplex, ℓ_1,ℓ_2 and ℓ_∞ balls, and the cones $\mathbb{R}^n_+, \mathbb{L}^n_+, \mathbb{S}^n_+$ and $K_{\exp}$ (as already mentioned, in these cases, the proximal operator is in fact a projection over a convex set). The proximal operator is also known, e.g., for the following functions:

- $f(\boldsymbol{x}) = \|\boldsymbol{x}\|_p$, for $p \in \{1, 2, \infty\}$.

- $f(\boldsymbol{x}) = \boldsymbol{x}^T Q \boldsymbol{x} + \boldsymbol{p}^T \boldsymbol{x}$, where $Q \succeq 0$ [convex quadratic]
- $f(\boldsymbol{x}) = \sum_i \max(x_i, 0)$ [Hinge loss]
- $f(\boldsymbol{x}) = \sum_i x_i \log x_i$ [Entropy]
- $f(\boldsymbol{x}, \boldsymbol{y}) = \sum_i x_i \log(x_i/y_i)$ [Kullback-Leibler divergence]
- $f(\boldsymbol{x}) = -\sum_i \log(x_i)$ [Logarithmic barrier]

To derive the formula of $\mathbf{prox}_f(\boldsymbol{x})$ for many of the above functions, we can use the following rule for separable sums:

$$\text{if } f(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n) = \sum_i f_i(\boldsymbol{x}_i), \quad \text{then } \mathbf{prox}_f(\boldsymbol{x}) = \begin{bmatrix} \mathbf{prox}_{f_1}(\boldsymbol{x}_1) \\ \vdots \\ \mathbf{prox}_{f_n}(\boldsymbol{x}_n) \end{bmatrix}$$

Hence, the derivarion of $\mathbf{prox}_f(\boldsymbol{x})$ is particularly easy when f is a separable sum of functions of one variable.

Example:
Let $f(\boldsymbol{x}) = \|\boldsymbol{x}\|_1 = \sum_i |x_i|$, and let $t > 0$. To compute the proximal operator of $t \cdot f$, it suffices to compute the proximal operator of the function of one variable $x \mapsto t|x|$.
So we must find the minimizer u^* of $h(u) = t|u| + \frac{1}{2}(x-u)^2$. From the expression of $\partial|\cdot|$ (see Example #1) and the previous theorem, this point must satisfy one of the following conditions:

$$\big(x - u^* = -t \quad \text{and} \quad u^* < 0\big) \quad \text{or} \quad \big(x - u^* \in [-t, t] \quad \text{and} \quad u^* = 0\big) \quad \text{or} \quad \big(x - u^* = t \quad \text{and} \quad u^* > 0\big).$$

This system can be solved as follows:

$$\mathbf{prox}_{x \mapsto t|x|}(x) = u^* = \mathcal{T}_t(x) := \begin{cases} x + t & \text{if } x < -t \\ 0 & \text{if } x \in [-t, t] \\ x - t & \text{if } x > t \end{cases} \quad = [|x| - t]_+ \operatorname{sign}(x), \qquad \boxed{\#2}$$

where $[u]_+$ is a shorthand notation for $\max(0, u)$ and $\operatorname{sign}(x)$ takes the value $-1, 0$, or 1 for negative, zero, and positive values of x, respectively. The function $\mathcal{T}_t$ is called the *soft thresholding operator* (at level t). Finally, we obtain the proximal operator of $t \cdot f$ by applying the the soft thresholding operator componentwise:

$$\mathbf{prox}_{tf}(\boldsymbol{x}) = \mathcal{T}_t(\boldsymbol{x})$$

where the (multivariate) soft thresholding operator $\mathcal{T}_t$ evaluated at $\boldsymbol{x}$ is the vector with components $\mathcal{T}_t(x_i) = [|x_i| - t]_+ \operatorname{sign}(x_i)$. In vector notation, we can write $\mathbf{prox}_{tf}(\boldsymbol{x}) = \mathcal{T}_t(\boldsymbol{x}) = [|\boldsymbol{x}| - t\mathbf{1}]_+ \odot \operatorname{sign}(\boldsymbol{x})$, where all operations are elementwise, and $\boldsymbol{u} \odot \boldsymbol{v}$ denotes the elementwise product, i.e., $(\boldsymbol{u} \odot \boldsymbol{v})_i = u_i v_i$.

3 The proximal gradient method

Throughout this section (and the next one), we consider a *composite convex model*

$$\underset{\boldsymbol{x} \in \mathbb{R}^n}{\textbf{minimize}} \; F(\boldsymbol{x}) := f(\boldsymbol{x}) + g(\boldsymbol{x}), \tag{P}$$

where:

- $f : \mathbb{R}^n \to \mathbb{R} \cup \{\infty\}$ is convex, continuously differentiable, and $\exists L \geq 0$:

$$\|\nabla f(\boldsymbol{x}) - \nabla f(\boldsymbol{y})\| \leq L \|\boldsymbol{x} - \boldsymbol{y}\|, \quad \forall \boldsymbol{x}, \boldsymbol{y} \in \mathbf{dom}\, f,$$

i.e., the gradient of f has Lipschitz constant $L \geq 0$: We say that f is *L-smooth.*

- $g : \mathbb{R}^n \to \mathbb{R} \cup \{\infty\}$ is a poper closed, nonsmooth convex function with a *cheap proximal operator*. In practice, this means that $\exists k \in \mathbb{N}$ such that we can compute $\mathbf{prox}_{tg}(\boldsymbol{x})$ in $O(n \log^k(n))$, for all $\boldsymbol{x} \in \mathbb{R}^n$ and $t \geq 0$.

This is a simple model, but it contains as special cases many basic problems encountered in machine learning or signal processing. In particular, the case $g = 0$ corresponds to an *unconstrained, smooth convex optimization problem*, and the case $g = I_C$ (for some "simple" convex set C with a cheap projection operator) corresponds to the *constrained optimization of a smooth convex function over C*.

> **Lemma 6.** *If f is L-smooth for some $L \geq 0$, then*
>
> $$f(\boldsymbol{y}) \leq f(\boldsymbol{x}) + \langle \nabla f(\boldsymbol{x}), \boldsymbol{y} - \boldsymbol{x} \rangle + \frac{L}{2}\|\boldsymbol{x} - \boldsymbol{y}\|^2, \quad \forall \boldsymbol{x}, \boldsymbol{y} \in \mathbf{dom}\, f.$$

Proof. From the fundamental theorem of calculus,

$$f(\boldsymbol{y}) = f(\boldsymbol{x}) + \int_{t=0}^{1} \langle \nabla f(\boldsymbol{x} + t(\boldsymbol{y} - \boldsymbol{x})), \boldsymbol{y} - \boldsymbol{x} \rangle \, dt.$$

Then, we can write

$$|f(\boldsymbol{y}) - f(\boldsymbol{x}) - \langle \nabla f(\boldsymbol{x}), \boldsymbol{y} - \boldsymbol{x} \rangle| = \left| \int_{t=0}^{1} \langle \nabla f(\boldsymbol{x} + t(\boldsymbol{y} - \boldsymbol{x})) - \nabla f(\boldsymbol{x}), \boldsymbol{y} - \boldsymbol{x} \rangle \, dt \right|,$$

and we can use the Cauchy-Schwarz inequality to bound the RHS by

$$\int_{t=0}^{1} \|\nabla f(\boldsymbol{x} + t(\boldsymbol{y} - \boldsymbol{x})) - \nabla f(\boldsymbol{x})\| \cdot \|\boldsymbol{y} - \boldsymbol{x}\| \, dt \leq \int_{t=0}^{1} Lt\|\boldsymbol{y} - \boldsymbol{x}\|^2 \, dt = \frac{L}{2}\|\boldsymbol{y} - \boldsymbol{x}\|^2.$$

□

Remark This inequality is essentially the reversed version of the inequality of Lemma 3: The strong convexity parameter ν gives a quadratic underestimator of a convex function, while the smoothness constant L gives a quadratic overestimator.

It can also be proved that the converse statement holds if f is convex: A convex function f is L-smooth iff the inequality of Lemma 6 holds for all $\boldsymbol{x}, \boldsymbol{y} \in \mathbf{dom}\, f$. Moreover, if f is twice differentiable, the best constant is $L = \sup_{\boldsymbol{x}} \lambda_{\max} \nabla^2 f(\boldsymbol{x})$.

The basic idea of the proximal gradient method is to minimize a quadratic overestimator of $F = f + g$ at each iteration. We do this by taking an overestimator of f, but we keep the exact expression of g. Then, the minimization of the this overestimator reduces to evaluating the prox-operator of (a scaling of) g.

Formally, given our current iterate $\boldsymbol{x}^{(k)} \in \mathbf{dom}\, f$ and a step size $0 < t_k \leq \frac{1}{L}$, the function

$$\hat{F}(\boldsymbol{y}) = f(\boldsymbol{x}^{(k)}) + \langle \nabla f(\boldsymbol{x}^{(k)}), \boldsymbol{y} - \boldsymbol{x}^{(k)} \rangle + \frac{1}{2t_k}\|\boldsymbol{y} - \boldsymbol{x}^{(k)}\|^2 + g(\boldsymbol{y})$$

is an overestimator of $F(\boldsymbol{y})$. So, the next iterate is determined by computing

$$\begin{aligned}
\boldsymbol{x}^{(k+1)} :=& \operatorname*{argmin}_{\boldsymbol{y}} \ \hat{F}(\boldsymbol{y})\\
=& \operatorname*{argmin}_{\boldsymbol{y}} \ g(\boldsymbol{y}) + \boldsymbol{y}^T \nabla f(\boldsymbol{x}^{(k)}) + \frac{1}{2t_k}\|\boldsymbol{y} - \boldsymbol{x}^{(k)}\|^2\\
=& \operatorname*{argmin}_{\boldsymbol{y}} \ g(\boldsymbol{y}) + \boldsymbol{y}^T \nabla f(\boldsymbol{x}^{(k)}) + \frac{1}{2t_k}(\|\boldsymbol{y}\|^2 - 2\boldsymbol{y}^T\boldsymbol{x}^{(k)})\\
=& \operatorname*{argmin}_{\boldsymbol{y}} \ t_k\ g(\boldsymbol{y}) + \frac{1}{2}\|\boldsymbol{y}\|^2 - \boldsymbol{y}^T(\boldsymbol{x}^{(k)} - t_k\nabla f(\boldsymbol{x}^{(k)}))\\
=& \operatorname*{argmin}_{\boldsymbol{y}} \ t_k\ g(\boldsymbol{y}) + \frac{1}{2}\|\boldsymbol{y} - (\boldsymbol{x}^{(k)} - t_k\nabla f(\boldsymbol{x}^{(k)}))\|^2\\
=& \mathbf{prox}_{t_k g}(\boldsymbol{x}^{(k)} - t_k\nabla f(\boldsymbol{x}^{(k)}))
\end{aligned}$$

This is what we call the *Proximal Gradient Iteration*:

$$\boldsymbol{x}^{(k+1)} := \mathbf{prox}_{t_k g}(\boldsymbol{x}^{(k)} - t_k\nabla f(\boldsymbol{x}^{(k)}))$$

It should be observed that the above iteration generalizes the gradient method (which is obtained when $g = 0$, in which case the prox-operator is the identity), and the *projected gradient method* for constrained optimization over a convex set C (this method is obtained when g is an indicator funcion I_C; in this case, the proximal operator *projects* the temptative iterate $\boldsymbol{x}^{(k)} - t_k\nabla f(\boldsymbol{x}^{(k)})$ onto the feasible set C).

The analysis of the proximal gradient method relies on the following theorem:

Theorem 7. *Let $\boldsymbol{x} \in \mathbf{int\,dom}\, f$ denote the current iterate, and $\boldsymbol{x}^+$ be the* next *iterate, obtained after a step of size $t > 0$, i.e., $\boldsymbol{x}^+ = \mathbf{prox}_{tg}(\boldsymbol{x} - t\nabla f(\boldsymbol{x}))$. If the new iterate satisfies*

$$f(\boldsymbol{x}^+) \leq f(\boldsymbol{x}) + \nabla f(\boldsymbol{x})^T(\boldsymbol{x}^+ - \boldsymbol{x}) + \frac{1}{2t}\|\boldsymbol{x}^+ - \boldsymbol{x}\|^2, \tag{2}$$

(note that by Lemma 6, the above is always satisfied if $t \leq \frac{1}{L}$), then for all $\xi \in \mathbb{R}^n$ it holds:

$$F(\xi) - F(\boldsymbol{x}^+) \geq \frac{1}{2t}\big(\|\xi - \boldsymbol{x}^+\|^2 - \|\xi - \boldsymbol{x}\|^2\big).$$

Proof. We have $\boldsymbol{x}^+ = \mathbf{prox}_{tg}(\boldsymbol{x} - t\nabla f(\boldsymbol{x}))$, so it follows from Theorem 5 that $\boldsymbol{x} - t\nabla f(\boldsymbol{x}) - \boldsymbol{x}^+ \in \partial(tg)(\boldsymbol{x}^+)$. Hence, for all $\xi \in \mathbb{R}^n$,

$$\begin{aligned}
& tg(\xi) \geq tg(\boldsymbol{x}^+) + \langle \boldsymbol{x} - t\nabla f(\boldsymbol{x}) - \boldsymbol{x}^+, \xi - \boldsymbol{x}^+\rangle\\
\iff\ & g(\xi) - g(\boldsymbol{x}^+) \geq \frac{1}{t}\langle \boldsymbol{x} - \boldsymbol{x}^+, \xi - \boldsymbol{x}^+\rangle - \nabla f(\boldsymbol{x})^T(\xi - \boldsymbol{x}^+).
\end{aligned}$$

On the other hand,

$$\begin{aligned}
f(\xi) - f(\boldsymbol{x}^+) &= f(\xi) - f(\boldsymbol{x}) + f(\boldsymbol{x}) - f(\boldsymbol{x}^+)\\
&\geq f(\xi) - f(\boldsymbol{x}) - \nabla f(\boldsymbol{x})^T(\boldsymbol{x}^+ - \boldsymbol{x}) - \frac{1}{2t}\|\boldsymbol{x}^+ - \boldsymbol{x}\|^2
\end{aligned}$$

Summing the above inequalities, we obtain

$$F(\xi) - F(\boldsymbol{x}^+) \geq \underbrace{f(\xi) - f(\boldsymbol{x}) - \nabla f(\boldsymbol{x})^T(\xi - \boldsymbol{x})}_{\epsilon_f(\boldsymbol{x},\xi)} + \frac{1}{t}\langle \boldsymbol{x} - \boldsymbol{x}^+, \xi - \boldsymbol{x}^+\rangle - \frac{1}{2t}\|\boldsymbol{x}^+ - \boldsymbol{x}\|^2. \tag{3}$$

In the above expression, $\epsilon_f(\boldsymbol{x},\xi)$ represents the error between $f(\xi)$ and the first order approximation $f(\boldsymbol{x})+\nabla f(\boldsymbol{x})^T(\xi-\boldsymbol{x})$, which must be nonnegative by convexity of f. Finally, we obtain

$$F(\xi)-F(\boldsymbol{x}^+)\geq\frac{1}{2t}\left(2\langle\boldsymbol{x}-\boldsymbol{x}^+,\xi-\boldsymbol{x}^+\rangle-\|\boldsymbol{x}^+-\boldsymbol{x}\|^2\right),$$

and simple calculus shows that this lower bound coincides with the desired bound, $\frac{1}{2t}\big(\|\xi-\boldsymbol{x}^+\|^2-\|\xi-\boldsymbol{x}\|^2\big)$. □

An immediate corollary of this theorem (obtained by setting $\xi=\boldsymbol{x}$) is a bound on the function's decrement between two successive iterations:

> **Corollary 8.** *If the step size t is chosen such that Equation* (2) *holds, then*
>
> $$F(\boldsymbol{x})-F(\boldsymbol{x}^+)\geq\frac{1}{2t}\|\boldsymbol{x}-\boldsymbol{x}^+\|^2.$$

We next present an analysis of the proximal gradient method, *for the case where the Lipschitz constant L is known, and constant step sizes are used: $t_k=\frac{1}{L},\forall k\in\mathbb{N}$.* If L is not known, a common technique is to use a backtracking line search in order to find step sizes satisfying Equation (2); our analysis can easily be adapted to handle this case, cf. [3].

> **Theorem 9.** *We consider the proximal gradient method with $\boldsymbol{x}^{(0)}\in\mathbf{int\,dom}\,f$ and constant step sizes:*
>
> $$\boldsymbol{x}^{(k+1)}:=\mathbf{prox}_{\frac{1}{L}g}(\boldsymbol{x}^{(k)}-\frac{1}{L}\nabla f(\boldsymbol{x}^{(k)})).$$
>
> *Then, for any optimal solution $\boldsymbol{x}^*$ of Problem* (P)*,*
>
> $$F(\boldsymbol{x}^{(k)})-F(\boldsymbol{x}^*)\leq\frac{L}{2k}\|\boldsymbol{x}^{(0)}-\boldsymbol{x}^*\|^2,\qquad\forall k\geq 1.$$

Remark It can also be shown that the sequence $(\boldsymbol{x}^{(k)})_{k\in\mathbb{N}}$ converges to one solution of Problem (P).

Proof. Let $i\in\mathbb{N}$. By Theorem 7 at $\xi=\boldsymbol{x}^*$,

$$F(\boldsymbol{x}^*)-F(\boldsymbol{x}^{(i+1)})\geq\frac{L}{2}\big(\|\boldsymbol{x}^*-\boldsymbol{x}^{(i+1)}\|^2-\|\boldsymbol{x}^*-\boldsymbol{x}^{(i)}\|^2\big).$$

Summing over $i=0,\ldots,k-1$,

$$\begin{aligned}k\,F(\boldsymbol{x}^*)-\sum_{i=0}^{k-1}F(\boldsymbol{x}^{(i+1)})\geq\frac{L}{2}\big(\|\boldsymbol{x}^*-\boldsymbol{x}^{(k)}\|^2-\|\boldsymbol{x}^*-\boldsymbol{x}^{(0)}\|^2\big)&\geq-\frac{L}{2}\|\boldsymbol{x}^*-\boldsymbol{x}^{(0)}\|^2\\ \iff\sum_{i=1}^{k}F(\boldsymbol{x}^{(i)})-k\,F(\boldsymbol{x}^*)&\leq\frac{L}{2}\|\boldsymbol{x}^*-\boldsymbol{x}^{(0)}\|^2.\end{aligned}\tag{4}$$

We know from Corrolary 8 that the proximal gradient method is a *descent method*, i.e.,

$$F(\boldsymbol{x}^{(0)})\geq F(\boldsymbol{x}^{(1)})\geq F(\boldsymbol{x}^{(2)})\geq\ldots$$

Therefore, we have $\sum_{i=1}^k F(\boldsymbol{x}^{(i)})\geq k\,F(\boldsymbol{x}^{(k)})$. Combining this inequality with (4) yields

$$k\,F(\boldsymbol{x}^{(k)})-k\,F(\boldsymbol{x}^*)\leq\frac{L}{2}\|\boldsymbol{x}^*-\boldsymbol{x}^{(0)}\|^2\iff F(\boldsymbol{x}^{(k)})-F(\boldsymbol{x}^*)\leq\frac{L}{2k}\|\boldsymbol{x}^{(0)}-\boldsymbol{x}^*\|^2.$$

□

If we have an upper bound R on the distance between $\boldsymbol{x}^{(0)}$ and any optimal solution $\boldsymbol{x}^*$, the above theorem guarantees that the proximal gradient method finds an ϵ-suboptimal solution after $k \leq \lceil \frac{LR^2}{\epsilon} \rceil$ iterations. This cannot be considered as a polynomial algorithm, since ϵ is typically *part of the input of the problem.* If we are interested in finding a solution with n accurate digits in the objective value (i.e., $\epsilon = 10^{-n}$), then we need $O(10^n)$ iterations, which is exponential in the numer of bits required to store ϵ.

A much better convergence result can be achieved when the function f is $\nu-$strongly convex. In that case, we obtain a polynomial-time algorithm (*linear convergence rate*):

> **Theorem 10.** *If f is ν-strongly convex, then the proximal gradient method with constant step sizes ($t_k = \frac{1}{L}$) generates a sequence of points satisfying*
>
> *(i)* $\|\boldsymbol{x}^{(k)} - \boldsymbol{x}^*\|^2 \leq \left(1 - \frac{\nu}{L}\right)^k \|\boldsymbol{x}^{(0)} - \boldsymbol{x}^*\|^2$*;*
>
> *(ii)* $F(\boldsymbol{x}^{(k)}) - F(\boldsymbol{x}^*) \leq \frac{L}{2}\left(1 - \frac{\nu}{L}\right)^k \|\boldsymbol{x}^{(0)} - \boldsymbol{x}^*\|^2$,
>
> *where $\boldsymbol{x}^*$ denotes the* unique *optimal solution to Problem P. Consequently, an $\epsilon-$suboptimal solution is found after $k = \left\lceil \frac{\log(LR^2/2\epsilon)}{\log L/(L-\nu)} \right\rceil = O\left(\frac{L}{\nu}\log(LR^2/\epsilon)\right)$ iterations, where R is an upper bound on $\|\boldsymbol{x}^{(0)} - \boldsymbol{x}^*\|$.*

Proof. (Sketch). As f is strongly convex, we can refine the bound of Theorem 7, by noting that $\epsilon_f(\boldsymbol{x}, \xi) \geq \frac{\nu}{2}\|\xi - \boldsymbol{x}\|^2$ in Eq. (3) (this follows from Lemma 3). Applied to the points $\xi = \boldsymbol{x}^*$ and $\boldsymbol{x} = \boldsymbol{x}^{(i)}$, the refined bound gives

$$\begin{aligned} F(\boldsymbol{x}^*) - F(\boldsymbol{x}^{(i+1)}) &\geq \frac{L}{2}\left(\|\boldsymbol{x}^* - \boldsymbol{x}^{(i+1)}\|^2 - \|\boldsymbol{x}^* - \boldsymbol{x}^{(i)}\|^2\right) + \frac{\nu}{2}\|\boldsymbol{x}^* - \boldsymbol{x}^{(i)}\|^2 \\ &= \frac{L}{2}\|\boldsymbol{x}^* - \boldsymbol{x}^{(i+1)}\|^2 - \frac{L-\nu}{2}\|\boldsymbol{x}^* - \boldsymbol{x}^{(i)}\|^2. \end{aligned}$$

Now, we know that $F(\boldsymbol{x}^*) - F(\boldsymbol{x}^{(i+1)}) \leq 0$, so we obtain

$$\frac{L}{2}\|\boldsymbol{x}^* - \boldsymbol{x}^{(i+1)}\|^2 \leq \frac{L-\nu}{2}\|\boldsymbol{x}^* - \boldsymbol{x}^{(i)}\|^2 \iff \|\boldsymbol{x}^* - \boldsymbol{x}^{(i+1)}\|^2 \leq \left(1 - \frac{\nu}{L}\right)\|\boldsymbol{x}^* - \boldsymbol{x}^{(i)}\|^2.$$

Then, the first statement of the proof is obtained by elementary induction on k. For the second statement, we rewrite the second inequality of this proof as

$$\begin{aligned} F(\boldsymbol{x}^{(k)}) - F(\boldsymbol{x}^*) \leq \frac{L-\nu}{2}\|\boldsymbol{x}^* - \boldsymbol{x}^{(k-1)}\|^2 - \frac{L}{2}\|\boldsymbol{x}^* - \boldsymbol{x}^{(k)}\|^2 &\leq \frac{L-\nu}{2}\|\boldsymbol{x}^* - \boldsymbol{x}^{(k-1)}\|^2 \\ &\leq \frac{L-\nu}{2}\left(1 - \frac{\nu}{L}\right)^{k-1}\|\boldsymbol{x}^* - \boldsymbol{x}^{(0)}\|^2 \\ &= \frac{L}{2}\left(1 - \frac{\nu}{L}\right)^k \|\boldsymbol{x}^{(0)} - \boldsymbol{x}^*\|^2. \end{aligned}$$

□

4 The FISTA accelarated method

Accelerated gradient methods were discovered in the 80's by Nesterov [4], and allow to improve the convergence rate of the gradient method from $O(\frac{1}{k})$ to $O(\frac{1}{k^2})$. The technique was generalized by Beck and Teboulle [5] in 2009 to handle composite models like (P) with a non-smooth (but *proximable*) part. The resulting algorithm was called FISTA by its authors: the name comes from "Fast iterative shrinkage-thresholding algorithm", and describes the proximal gradient steps in the case of a lasso-penalty, $g(\boldsymbol{x}) = \|\boldsymbol{x}\|_1$.

The idea is to take a proximal gradient step from a point $\boldsymbol{y}^{(k)}$, i.e., $\boldsymbol{x}^{(k+1)} := \mathbf{prox}_{t_k g}(\boldsymbol{y}^{(k)} - t_k \nabla f(\boldsymbol{y}^{(k)}))$, where the point $\boldsymbol{y}^{(k)}$ is a (well chosen) linear combination of the two previous iterates $\boldsymbol{x}^{(k)}$ and $\boldsymbol{x}^{(k-1)}$. We next present the FISTA method for the case of constant step sizes, $t_k = \frac{1}{L}, \forall k$. As in the previous section, we point out that the analysis can be adapted to handle backtracking line search.

```
FISTA
Initialization: $\boldsymbol{y}^{(0)} = \boldsymbol{x}^{(0)} \in \textbf{int}\,\textbf{dom}\, f$, $\tau_0 = 1$.
For $k = 0, 1, 2, \ldots$,
    1. $\boldsymbol{x}^{(k+1)} = \textbf{prox}_{\frac{1}{L}g}(\boldsymbol{y}^{(k)} - \frac{1}{L}\nabla f(\boldsymbol{y}^{(k)}))$
    2. $\tau_{k+1} = \frac{1+\sqrt{1+4\tau_k^2}}{2}$
    3. $\boldsymbol{y}^{(k+1)} = \boldsymbol{x}^{(k+1)} + \left(\frac{\tau_k - 1}{\tau_{k+1}}\right)(\boldsymbol{x}^{(k+1)} - \boldsymbol{x}^{(k)})$
```

Note The funny formula for the sequence τ_k actually corresponds to the positive root of the equation $\tau_{k+1}^2 - \tau_{k+1} = \tau_k^2$. An easy induction shows that

$$\tau_k \geq \frac{k+2}{2} \geq 1, \quad \forall k. \tag{5}$$

Theorem 11. *Consider the sequence of iterates $\boldsymbol{x}^{(k)}$ generated by FISTA (with constant step sizes $t_k = \frac{1}{L}, \forall k$). Then, for any optimal solution $\boldsymbol{x}^*$ to Problem* (P)*, it holds*

$$F(\boldsymbol{x}^{(k)}) - F(\boldsymbol{x}^*) \leq \frac{2L\|\boldsymbol{x}^{(0)} - \boldsymbol{x}^*\|^2}{(k+1)^2}.$$

Proof. Let $k \geq 1$. We introduce the notation

$$\delta_k = F(\boldsymbol{x}^{(k)}) - F(\boldsymbol{x}^*) \quad \text{and} \quad \boldsymbol{u}^{(k)} = (\tau_{k-1} - 1)\boldsymbol{x}^{(k-1)} + \boldsymbol{x}^* - \tau_{k-1}\boldsymbol{x}^{(k)}, \quad \forall k.$$

We will also need the point $\xi = \frac{\boldsymbol{x}^*}{\tau_k} + (1 - \frac{1}{\tau_k})\boldsymbol{x}^{(k)}$, which satisfies

$$\tau_k(\xi - \boldsymbol{x}^{(k+1)}) = \boldsymbol{x}^* + (\tau_k - 1)\boldsymbol{x}^{(k)} - \tau_k\boldsymbol{x}^{(k+1)} = \boldsymbol{u}^{(k+1)}. \tag{6}$$

By definition, we have $\boldsymbol{y}^{(k)} = \boldsymbol{x}^{(k)} + \frac{\tau_{k-1}-1}{\tau_k}(\boldsymbol{x}^{(k)} - \boldsymbol{x}^{(k-1)})$, i.e., $\tau_k(\boldsymbol{x}^{(k)} - \boldsymbol{y}^{(k)}) = (\tau_{k-1} - 1)(\boldsymbol{x}^{(k-1)} - \boldsymbol{x}^{(k)})$. Therefore,

$$\begin{aligned}
\tau_k(\xi - \boldsymbol{y}^{(k)}) &= \boldsymbol{x}^* + (\tau_k - 1)\boldsymbol{x}^{(k)} - \tau_k\boldsymbol{y}^{(k)} \\
&= \boldsymbol{x}^* - \boldsymbol{x}^{(k)} + \tau_k(\boldsymbol{x}^{(k)} - \boldsymbol{y}^{(k)}) \\
&= \boldsymbol{x}^* - \boldsymbol{x}^{(k)} + (\tau_{k-1} - 1)(\boldsymbol{x}^{(k-1)} - \boldsymbol{x}^{(k)}) \\
&= \boldsymbol{x}^* + (\tau_{k-1} - 1)\boldsymbol{x}^{(k-1)} - \tau_{k-1}\boldsymbol{x}^{(k)} \\
&= \boldsymbol{u}^{(k)}
\end{aligned} \tag{7}$$

We apply Theorem 7 to the point ξ, when the current iterate is $\boldsymbol{x} = \boldsymbol{y}^{(k)}$. Note that the proximal step evaluated at $\boldsymbol{y}^{(k)}$ yields the new iterate $\boldsymbol{x}^+ = \boldsymbol{x}^{(k+1)}$, so the theorem simply gives

$$F(\xi) - F(\boldsymbol{x}^{(k+1)}) \geq \frac{L}{2}\big(\|\xi - \boldsymbol{x}^{(k+1)}\|^2 - \|\xi - \boldsymbol{y}^{(k)}\|^2\big) = \frac{L}{2\tau_k^2}\big(\|\boldsymbol{u}^{(k+1)}\|^2 - \|\boldsymbol{u}^{(k)}\|^2\big).$$

Now, we use the convexity of F. Since $\tau_k \geq 1$, the point ξ is a convex combination of $\boldsymbol{x}^{(k)}$ ans $\boldsymbol{x}^*$, and it holds

$$F(\xi) - F(\boldsymbol{x}^{(k+1)}) \leq \frac{1}{\tau_k}F(\boldsymbol{x}^*) + (1 - \frac{1}{\tau_k})F(\boldsymbol{x}^{(k)}) - F(\boldsymbol{x}^{(k+1)}) = (1 - \frac{1}{\tau_k})\delta_k - \delta_{k+1}.$$

Combining the above two inequalities yields

$$(1-\frac{1}{\tau_k})\delta_k - \delta_{k+1} \geq \frac{L}{2\tau_k^2}\big(\|\boldsymbol{u}^{(k+1)}\|^2 - \|\boldsymbol{u}^{(k)}\|^2\big) \iff \frac{2}{L}[(\tau_k^2-\tau_k)\delta_k - \tau_k^2\delta_{k+1}] \geq \|\boldsymbol{u}^{(k+1)}\|^2 - \|\boldsymbol{u}^{(k)}\|^2.$$

By construction of the sequence (τ_k), we have $\tau_k^2 - \tau_k = (\tau_{k-1})^2$. We have thus shown that

$$\frac{2}{L}[(\tau_{k-1})^2\delta_k - \tau_k^2\delta_{k+1}] \geq \|\boldsymbol{u}^{(k+1)}\|^2 - \|\boldsymbol{u}^{(k)}\|^2.$$

The above reasonning was for an arbitrary index $k \geq 1$. So it holds

$$\|\boldsymbol{u}^{(k+1)}\|^2 + \frac{2}{L}\tau_k^2\delta_{k+1} \leq \|\boldsymbol{u}^{(k)}\|^2 + \frac{2}{L}\tau_{k-1}^2\delta_k, \quad \forall k \geq 1.$$

This means that the sequence $v_k = \|\boldsymbol{u}^{(k)}\|^2 + \frac{2}{L}\tau_{k-1}^2\delta_k$ is nonincreasing, so we can write

$$\frac{2}{L}\tau_{k-1}^2\delta_k \leq \|\boldsymbol{u}^{(k)}\|^2 + \frac{2}{L}\tau_{k-1}^2\delta_k \leq \|\boldsymbol{u}^{(1)}\|^2 + \frac{2}{L}\tau_0^2\delta_1 = \|\boldsymbol{x}^* - \boldsymbol{x}^{(1)}\|^2 + \frac{2}{L}\delta_1.$$

To conclude the proof, we apply Theorem 7 to the point $\boldsymbol{x}^*$, when the proximal step is evaluated at $\boldsymbol{x} = \boldsymbol{y}^{(0)}$, yielding the next iterate $\boldsymbol{x}^+ = \boldsymbol{x}^{(1)}$:

$$F(\boldsymbol{x}^*) - F(\boldsymbol{x}^{(1)}) \geq \frac{L}{2}\big(\|\boldsymbol{x}^* - \boldsymbol{x}^{(1)}\|^2 - \|\boldsymbol{x}^* - \boldsymbol{x}^{(0)}\|^2\big) \iff \|\boldsymbol{x}^* - \boldsymbol{x}^{(1)}\|^2 + \frac{2}{L}\delta_1 \leq \|\boldsymbol{x}^* - \boldsymbol{x}^{(0)}\|^2.$$

Finally, we obtain the desired result by combining the last 2 inequalities and $\tau_{k-1} \geq (k+1)/2$ (see eq. (5)):

$$\frac{2}{L}\tau_{k-1}^2\delta_k \leq \|\boldsymbol{x}^* - \boldsymbol{x}^{(0)}\|^2 \iff \delta_k \leq \frac{2L\|\boldsymbol{x}^* - \boldsymbol{x}^{(0)}\|^2}{(k+1)^2},$$

□

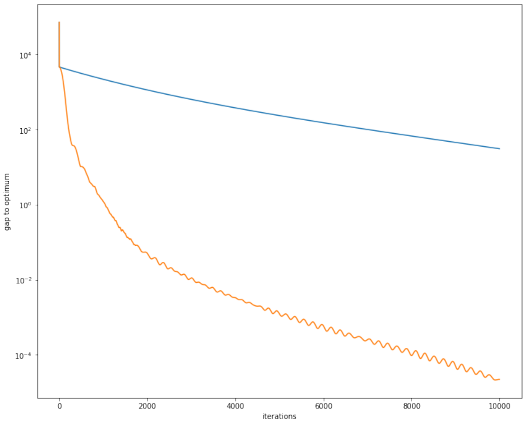
Example:
The following plot shows the evolution of the gap to optimality $\delta_k = F(\boldsymbol{x}^{(k)}) - F(\boldsymbol{x}^*)$, for the standard proximal gradient algorithm (upper curve), and the accelerated FISTA method (lower curve). The problem solved is a Lasso regression problem:

$$\underset{\boldsymbol{x}\in\mathbb{R}^n}{\textbf{minimize}}\ \underbrace{\lambda\|A\boldsymbol{x}-\boldsymbol{y}\|^2}_{f(\boldsymbol{x})} + \underbrace{\|\boldsymbol{x}\|_1}_{g(\boldsymbol{x})}.$$

The data $(A, \boldsymbol{y})$ was randomly generated, with components of $A \in \mathbb{R}^{5000\times 1000}$ drawn independently at random in $[0, 1]$, and $\boldsymbol{y}$ was set to $A\boldsymbol{x}_0 + \boldsymbol{\epsilon}$ for a *sparse* vector $\boldsymbol{x}_0$ and a vector of noise $\boldsymbol{\epsilon}$.

In this example, the gradient of f is $\nabla f(\boldsymbol{x}) = 2\lambda A^T(A\boldsymbol{x}-\boldsymbol{y})$, so f is L-smooth for $L = 2\lambda \cdot \lambda_{\max}(A^TA)$. The proximity operator of g is the soft thresholding operator, cf. Example #2. Therefore, with constant stepsizes $t_k = \frac{1}{L}$, the proximal steps take the form

$$\boldsymbol{x}^+ \leftarrow \mathcal{T}_{\frac{1}{L}}\left(\boldsymbol{x} - \frac{2\lambda}{L}A^T(A\boldsymbol{x}-\boldsymbol{y})\right).$$



#3

- On this example, FISTA converges much more quickly. The total time to run $k = 10^4$ iterations was 45 s. for the proximal gradient algorithm, and about 60s. for FISTA. Although FISTA iterations are a bit more expensive, the convergence is several orders of magnitude faster.
- We see on the picture that FISTA *is not* a descent method, the function value shows typical oscillations.
- The bounds given by Theorems 9 and 11 are pessimistic. After $k = 10^4$ iterations, the proximal gradient method has a gap of $\delta_k = 30.99 \ll \frac{LR^2}{2k} = 1397.7$, and FISTA has a gap of $\delta_k = 2.22 \cdot 10^{-5} \ll \frac{2LR^2}{(k+1)^2} = 0.558$.
- In that case, the function f is ν-strongly convex for $\nu = 2\lambda \cdot \lambda_{\min}(A^TA)$, but the ratio between L and ν is so huge that the bound $(1 - \frac{\nu}{L})$ on the convergence rate is useless for a reasonable number of iterations. After $k = 10^4$ iterations, Theorem 10 gives the bound $\delta \leq \frac{L}{2}(1-\nu/L)^k R^2 = 5 \cdot 10^6$. This bound becomes better than $\frac{LR^2}{2k}$ after about 55.000 iterations, and better than $\frac{2LR^2}{(k+1)^2}$ after 170.000 iterations.

5 Optimality of accelerated (proximal) gradient methods

We conclude this chapter with a beautiful result, which shows that the accelerated gradient descent (and hence its proximal gradient version FISTA) are essentially optimal among the class of first order methods: Unless we know that the function to minimize has a special property (such as ν-strong convexity), no first-order algorithm can guarantee a convergence better than $O(LR^2/(k+1)^2)$ in the worst-case.

> **Theorem 12.** *There exists a function $f : \mathbb{R}^{2k+1} \to \mathbb{R}$ which is twice differentiable and L-smooth, such that for any sequence $(\boldsymbol{x}^{(i)})_{i\in\mathbb{N}}$ satisfying $\boldsymbol{x}^{(i+1)} \in \boldsymbol{x}^{(0)} + \operatorname{span}(\nabla f(\boldsymbol{x}^{(0)}), \ldots, \nabla f(\boldsymbol{x}^{(i)}))$, $\forall i \in \mathbb{N}$, it holds*
>
> $$f(\boldsymbol{x}^{(k)}) - f(\boldsymbol{x}^*) \geq \frac{3L\|\boldsymbol{x}^{(0)} - \boldsymbol{x}^*\|^2}{32(k+1)^2}.$$

Proof. For all $k \in \mathbb{N}$, define $f_k(\boldsymbol{x}) = \frac{L}{4}(\frac{1}{2}\boldsymbol{x}^T A\boldsymbol{x} - \boldsymbol{e}_1^T\boldsymbol{x})$, where

$$A = \begin{pmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 2 \end{pmatrix} \in \mathbb{S}^k.$$

One can show that $A \succeq 0$, and $\lambda_{\max}(A) \leq 4$. Hence, f_k is convex and the Lipschitz constant of its gradient is $\frac{L}{4}\lambda_{\max}(A) \leq L$.

The function f_k is minimized over $\mathbb{R}^k$ at $\boldsymbol{x}^* = A^{-1}\boldsymbol{e}_1$, and we can show that $(\boldsymbol{x}^*)_i = 1 - \frac{i}{k+1}$, for $i = 1, \ldots, k$. Hence, its minimum is $f_k(\boldsymbol{x}^*) = \frac{L}{4}(\frac{1}{2}\boldsymbol{x}^{*T}\boldsymbol{e}_1 - \boldsymbol{x}^{*T}\boldsymbol{e}_1) = -\frac{L}{8}(1 - \frac{1}{k+1})$.

Now, let $f = f_{2k+1}$. Let us assume (without loss of generality) that $\boldsymbol{x}^{(0)} = \boldsymbol{0}$, and that $\boldsymbol{x}^{(i+1)} \in \operatorname{span}(\nabla f(\boldsymbol{x}^{(0)}), \ldots, \nabla f(\boldsymbol{x}^{(i)}))$, $\forall i$. Then, a simple induction shows that for all $i < 2k+1$,

$$\operatorname{span}(\nabla f(\boldsymbol{x}^{(0)}), \ldots, \nabla f(\boldsymbol{x}^{(i)})) \subseteq \operatorname{span}(\boldsymbol{e}_1, \ldots, \boldsymbol{e}_{i+1}).$$

Indeed, we have $\nabla f(\boldsymbol{x}^{(0)}) = \frac{L}{4}(A\boldsymbol{x}^{(0)} - \boldsymbol{e}_1) = -\frac{L}{4}\boldsymbol{e}_1$. Then, assuming the induction hypothesis is true for $i = j-1$, we have $\boldsymbol{x}_j \in \operatorname{span}(\nabla f(\boldsymbol{x}^{(0)}), \ldots, \nabla f(\boldsymbol{x}^{(j-1)})) = \operatorname{span}(\boldsymbol{e}_1, \ldots, \boldsymbol{e}_j)$, i.e., $\boldsymbol{x}_j$ has nonzero components on its first j coordinates only. So $\nabla f(\boldsymbol{x}^{(j)}) = \frac{L}{4}(A\boldsymbol{x}^{(j)} - \boldsymbol{e}_1)$ has nonzero components on its first $j+1$ coordinates only, and $\operatorname{span}(\nabla f(\boldsymbol{x}^{(0)}), \ldots, \nabla f(\boldsymbol{x}^{(j)})) = \operatorname{span}(\boldsymbol{e}_1, \ldots, \boldsymbol{e}_j, \nabla f(\boldsymbol{x}^{(j)})) \subseteq \operatorname{span}(\boldsymbol{e}_1, \ldots, \boldsymbol{e}_j, \boldsymbol{e}_{j+1})$.

To obtain the bound of the theorem, we observe that since $\boldsymbol{x}^{(k)} \in \operatorname{span}(\boldsymbol{e}_1, \ldots, \boldsymbol{e}_k)$, it holds $f(\boldsymbol{x}^{(k)}) = f_k(\hat{\boldsymbol{x}}^{(k)})$, where $\hat{\boldsymbol{x}}^{(k)}$ is the k-dimensional vector with the first k coordinates of $\boldsymbol{x}^{(k)} \in \mathbb{R}^{2k+1}$. Hence,

$$f(\boldsymbol{x}^{(k)}) \geq \inf_{\boldsymbol{x}} f_k(\boldsymbol{x}) = -\frac{L}{8}(1 - \frac{1}{k+1}).$$

We can now conclude:

$$\frac{f(\boldsymbol{x}^{(k)}) - f(\boldsymbol{x}^*)}{\|\boldsymbol{x}^{(0)} - \boldsymbol{x}^*\|^2} \geq \frac{-\frac{L}{8}(1 - \frac{1}{k+1}) + \frac{L}{8}(1 - \frac{1}{2k+2})}{\frac{2}{3}(k+1)} = \frac{L}{8 \cdot \frac{2}{3}(k+1)} \frac{1}{2k+2} = \frac{3L}{32(k+1)^2},$$

where in the first inequality we have used

$$\|\boldsymbol{x}^* - \boldsymbol{x}^{(0)}\|^2 = \|\boldsymbol{x}^*\|^2 = \sum_{i=1}^{2k+1}(1 - \frac{i}{2k+2})^2 = \frac{8k^2 + 10k + 3}{12(1+k)} \leq \frac{8(k+1)^2}{12(1+k)} = \frac{2}{3}(k+1).$$

□

References

[1] Candes, E.J., & Plan, Y. (2010). Matrix completion with noise. Proceedings of the IEEE, 98(6), 925–936. arXiv:0903.3131.

[2] `http://proximity-operator.net/index.html`, maintained by Chierchia, G., Chouzenoux, E., Combettes, P.L., & Pesquet, J.C.

[3] Beck, A. (2017). First-Order Methods in Optimization (Vol. 25). SIAM.

[4] Nesterov, Y.E. (1983). A method for solving the convex programming problem with convergence rate $O(1/k^2)$. In Dokl. Akad. Nauk SSSR (Vol. 269, pp. 543-547).

[5] Beck, A., & Teboulle, M. (2009). A fast iterative shrinkage-thresholding algorithm for linear inverse problems. SIAM journal on imaging sciences, 2(1), 183-202.