HAL
open science

# On the Effectiveness of Richardson Extrapolation in Data Science

Francis Bach

## To cite this version:

Francis Bach. On the Effectiveness of Richardson Extrapolation in Data Science. SIAM Journal on Mathematics of Data Science, 2021, 3 (4), pp.1251-1277. 10.1137/21M1397349 . hal-02470950v2

**HAL Id: hal-02470950**

**https://hal.science/hal-02470950v2**

Submitted on 9 Jul 2020

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# On the Effectiveness of Richardson Extrapolation in Machine Learning

Francis Bach
INRIA - Département d'Informatique de l'Ecole Normale Supérieure
PSL Research University Paris, France
francis.bach@inria.fr

July 9, 2020

**Abstract**

Richardson extrapolation is a classical technique from numerical analysis that can improve the approximation error of an estimation method by combining linearly several estimates obtained from different values of one of its hyperparameters, without the need to know in details the inner structure of the original estimation method. The main goal of this paper is to study when Richardson extrapolation can be used within machine learning, beyond the existing applications to step-size adaptations in stochastic gradient descent. We identify two situations where Richardson interpolation can be useful: (1) when the hyperparameter is the number of iterations of an existing iterative optimization algorithm, with applications to averaged gradient descent and Frank-Wolfe algorithms (where we obtain asymptotically rates of $O(1/k^2)$ on polytopes, where $k$ is the number of iterations), and (2) when it is a regularization parameter, with applications to Nesterov smoothing techniques for minimizing non-smooth functions (where we obtain asymptotically rates close to $O(1/k^2)$ for non-smooth functions), and ridge regression. In all these cases, we show that extrapolation techniques come with no significant loss in performance, but with sometimes strong gains, and we provide theoretical justifications based on asymptotic developments for such gains, as well as empirical illustrations on classical problems from machine learning.

## 1 Introduction

Many machine learning methods can be cast as looking for approximations of some ideal quantity which cannot be readily computed from the data at hand: this ideal quantity can be the predictor learned from infinite data, or an iterative algorithm run for infinitely many iterations. Taking theirs roots in optimization and more generally numerical analysis, many accelerations techniques have been developed to tighten these approximations with as few changes as possible to the original method.

While some acceleration techniques add some simple modifications to a known algorithm, such as Nesterov acceleration for the gradient descent method (Nesterov, 1983), *extrapolation* techniques are techniques that do not need to know the fine inner structure of the method to be accelerated. These methods are only based on the observations of solutions of the original method.

Extrapolation techniques work on the vector-valued output $x_t \in \mathbb{R}^d$ of the original method that depends on some controllable real-valued quantity $t$, which can be the number of iterations or some regularization parameter, and more generally any parameter that controls both the running time and the approximation error of the algorithm. When $t$ tends to $t_\infty$ (which is typically 0 or $+\infty$),

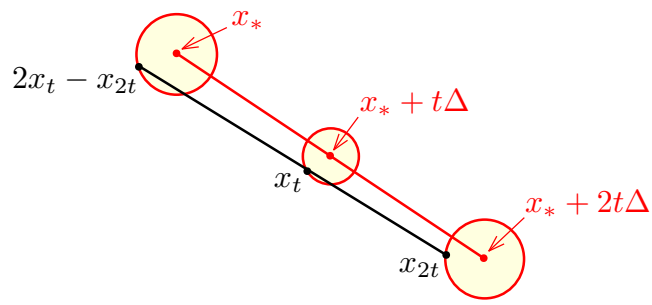


Figure 1: Illustration of Richardson extrapolation for $t_\infty = 0$ and $x_t = x_* + t\Delta + O(t^2)$. Iterates (in black) with their first-order expansions (in red). The deviations (represented by circles) are of order $O(t^2)$. Adapted from Dieuleveut et al. (2019).

we will assume that $x_t$ has an asymptotic expansion of the form

$$x_t = x_* + g_t + O(h_t),$$

where $x_*$ is the desired output, $g_t \in \mathbb{R}^d$ is the asymptotic equivalent of $x_t - x_*$, and $h_t = o(\|g_t\|)$. The key question in extrapolation is the following: from the knowledge of $x_t$ for potentially several $t$'s, how can we better approximate $x_*$, *without the full knowledge* of $g_t$?

For exponentially converging algorithms, there exist several "non-linear" schemes that combine linearly several values of $x_t$ with weights that depend non-linearly on the iterates, such as Aitken's $\Delta^2$ process (Aitken, 1927) or Anderson acceleration (Anderson, 1965), which has recently been shown to provide significant acceleration to linearly convergent gradient-based algorithms (Scieur et al., 2016). In this paper, we consider dependence in powers of $t$, where Richardson extrapolation excels (see, e.g., Richardson, 1911; Joyce, 1971; Gautschi, 1997).

We thus assume that

$$g_t = t^\alpha \cdot \Delta,$$

and $h_t = t^\beta$ is a power of $t$ such that $h_t = o(\|g_t\|)$, where $\alpha \in \mathbb{R}$ is known but $\Delta \in \mathbb{R}^d$ is unknown, that is

$$x_t = x_* + t^\alpha \Delta + O(t^\beta).$$

In all our cases, $\alpha = -1$ when $t_\infty = +\infty$ and $\alpha = 1$ when $t = 0$. Richardson extrapolation is simply combining two iterates with different values of $t$ so that the zero-th order term $x_*$ is preserved, while the first-order term cancels, for example:

$$\begin{aligned} & 2x_t - x_{2^{1/\alpha} t} \\ = \; & 2(x_* + t^\alpha \Delta + O(t^\beta)) - (x_* + 2t^\alpha \Delta + O(t^\beta)) \\ = \; & x_* + O(t^\beta). \end{aligned}$$

See an illustration in Figure 1 for $\alpha = 1$, $\beta = 2$, and $t_\infty = 0$. Note that: (a) the choice of $2^{1/\alpha} \neq 1$ as a multiplicative factor is arbitrary and chosen for its simplicity when $|\alpha| = 1$, and (b) Richardson extrapolation can be used with $m+1$ iterates to remove the first $m$ terms in a asymptotic expansion, where the powers of the expansion are known and not the associated vector-valued constants (see examples in Section 3).

The main goal of this paper is to study when Richardson extrapolation can be used within machine learning, beyond the existing explicit applications to step-size adaptations in stochastic gradient descent (Durmus et al., 2016; Dieuleveut et al., 2019), and implicit wide-spread use in "tail-averaging" (see more details in Section 2.1).

We identify two situations where Richardson interpolation can be useful:

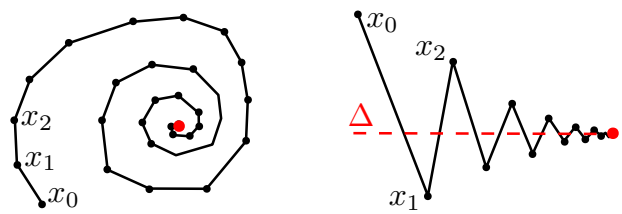


Figure 2: Left: Oscillating convergence, where Richardson extrapolation does not lead to any gain. Right: non-oscillating convergence, with a main direction $\Delta$ (in red dotted), where Richardson extrapolation can be beneficial if the oscillations orthogonal to the direction $\Delta$ are negligible compared to convergence along the direction $\Delta$.

- $t = k$ is the number of iterations of an existing iterative optimization algorithm converging to $x_*$, where then $\alpha = -1$ and $t_\infty = +\infty$, and Richardson extrapolation considers, for $k$ even, $x_k^{(1)} = 2x_k - x_{k/2}$. We consider in Section 2, averaged gradient descent and Frank-Wolfe algorithms (where we obtain asymptotically rates of $O(1/k^2)$ on polytopes, where $k$ is the number of iterations).
- $t = \lambda$ is a regularization parameter, where then $\alpha = 1$ and $t_\infty = 0$, and Richardson extrapolation considers $x_\lambda^{(1)} = 2x_\lambda - x_{2\lambda}$. We consider in Section 3, Nesterov smoothing techniques for minimizing non-smooth functions (where we obtain asymptotically rates close to $O(1/k^2)$ for non-smooth functions), and ridge regression (where we obtain estimators with lower bias).

As we will show, extrapolation techniques come with no significant loss in performance, but with sometimes strong gains, and the goal of this paper is to provide theoretical justifications for such gains, as well as empirical illustrations on classical problems from machine learning. Note that we aim for the simplest asymptotic results (most can be made non-asymptotic with extra assumptions).

## 2 Extrapolation on the Number of Iterations

In this section, we consider extrapolation based on the number of iterations $k$, that is, for the simplest case

$$x_k^{(1)} = 2x_k - x_{k/2},$$

for optimization algorithms aimed at minimizing on $\mathbb{R}^d$ If $x_k$ is converging to a minimizer $x_*$, then so is $x_{k/2}$, and thus also $x_k^{(1)} = 2x_k - x_{k/2}$; moreover, we have $\|x_k^{(1)} - x_*\|_2 \leqslant 2\|x_k - x_*\|_2 + \|x_{k/2} - x_*\|_2$, so even if there are no cancellations, performance is never significantly deteriorated (the risk is essentially to lose half of the iterations).

The potential gains depend on the way $x_k$ converges to $x_*$. The existence of a convergence rate of the form $f(x_k) - f(x_*) = O(1/k)$ or $O(1/k^2)$ is not enough, as Richardson extrapolation requires a specific direction of asymptotic convergence. As illustrated in Figure 2, some algorithms are oscillating around their solutions, while some converge with a specific direction. Only the latter ones can be accelerated with Richardson extrapolation, while the former ones are good candidates for Anderson acceleration (Anderson, 1965; Scieur et al., 2016).

We now consider three algorithms: (1) averaged gradient descent, where extrapolation is at its best, as it transforms an $O(1/t^2)$ convergence rate into an exponential one, (2) accelerated gradient descent, where extrapolation does not bring anything, and (3) Frank-Wolfe algorithms, where the situation is mixed (sometimes it helps, sometimes it does not).

## 2.1 Averaged gradient descent

We consider the usual gradient descent algorithm

$$x_k = x_{k-1} - \gamma f'(x_{k-1}),$$

where $\gamma \geqslant 0$ is a step-size, with Polyak-Ruppert averaging (Polyak and Juditsky, 1992; Ruppert, 1988):

$$\bar{x}_k = \frac{1}{k} \sum_{i=0}^{k-1} x_i.$$

Averaging is key to robustness to potential noise of the gradients (Polyak and Juditsky, 1992; Nemirovski et al., 2009). However it comes with the unintended consequence of losing the exponential forgetting of initial conditions for strongly convex problems (Bach and Moulines, 2011).

A common way to restore exponential convergence (up to the noise level in the stochastic case) is to consider "tail-averaging", that is, to replace $\bar{x}_k$ by the average of only the latest $k/2$ iterates (Jain et al., 2018). As shown below for $k$ even, this corresponds exactly to Richardson extrapolation:

$$\frac{2}{k} \sum_{i=k/2}^{k-1} x_i = \frac{2}{k} \sum_{i=0}^{k-1} x_i - \frac{2}{k} \sum_{i=0}^{k/2-1} x_i = 2\bar{x}_k - \bar{x}_{k/2}.$$

While Jain et al. (2018) focuses on a non-asymptotic analysis for stochastic problems for least-squares regression, we now provide an asymptotic analysis for general convex objective functions and non-stochastic problems (see proof in Appendix B).

**Proposition 1** *Assume $f$ convex, three-times differentiable with Hessian eigenvalues bounded by $L$, bounded third-order derivatives, and a unique minimizer $x_* \in \mathbb{R}^d$ such that $f''(x_*)$ is positive definite. If $\gamma \leqslant 1/L$, then*

$$\bar{x}_k = x_* + \frac{1}{k}\Delta + O(\exp(-k\lambda)),$$

*where $\Delta = \sum_{i=0}^{\infty}(x_i - x_*)$ and $\lambda$ is proportional to $\gamma\lambda_{\min}(f''(x_*))$.*

Note that: (a) before Richardson extrapolation, the asymptotic convergence rate will be of the order $O(1/k^2)$, which is better than the usual $O(1/k)$ upper-bound for the rate of gradient descent, but with a stronger assumption that in fact leads to exponential convergence before averaging, (b) while $\Delta$ has a simple expression, it cannot be computed in practice, (c) that Richardson extrapolation leads to an exponentially convergent algorithm from an algorithm converging asymptotically in $O(1/k^2)$ for functions values, and (d) that in the presence of noise in the gradients, the exponential convergence would only be up to the noise level. See Figure 3 (left plot) for an illustration with noisy gradients.

## 2.2 Accelerated gradient descent

In the section above, we considered averaged gradient descent, which is asymptotically converging as $O(1/k^2)$ and on which Richardson extrapolation could be used with strong gains. Is it possible also for the accelerated gradient descent (Nesterov, 1983), which has a (non-asymptotic) convergence rate of $O(1/k^2)$ for convex functions?

It turns out that the behavior of the iterates of accelerated gradient descent is exactly of the form depicted in the left plot of Figure 2: that is, the iterates $x_k$ oscillate around the optimum, as can be seen from the spectral analysis for quadratic problems, in continuous time (Su et al., 2016) or discrete time (Flammarion and Bach, 2015). Richardson extrapolation is of no help, but is not degrading performance too much. See Figure 3 (right plot) for an illustration.

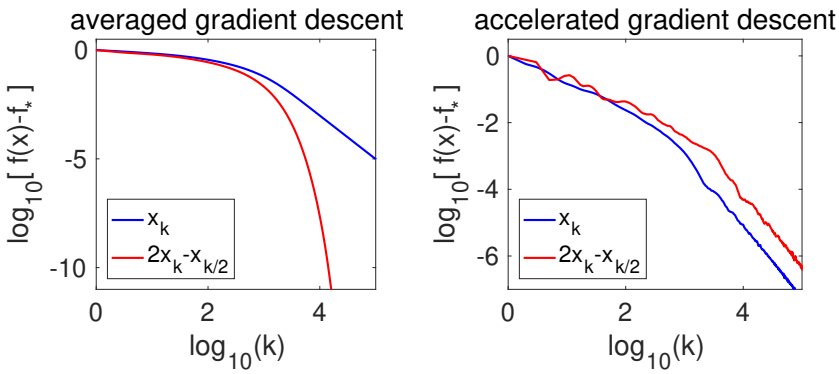


Figure 3: Left: averaged gradient descent on a logistic regression problem in dimension $d = 400$, and with $n = 4000$ observations $\min_{x \in \mathbb{R}^d} f(x) = \frac{1}{n} \sum_{i=1}^n \log(1 + \exp(-b_i x^\top a_i))$, with $(a_i, b_i) \in \mathbb{R}^d \times \{-1, 1\}$. The covariance matrix of (Gaussian) inputs has eigenvalues $1/j$, $j = 1, \ldots, d$; the lowest eigenvalue is $1/400$ and therefore we can see the effect of strong convexity starting between $k = 100$ and 1000 iterations; moreover, for the regular averaged recursion, the line in the log-log plot has slope $-2$. Right: accelerated gradient descent on a quadratic optimization problem in dimension $d = 1000$ and a Hessian whose eigenvalues are $1/j^2$, $j = 1, \ldots, d$; with such eigenvalues, the local linear convergence is not observed and we have a line of slope $-2$.

## 2.3 Frank-Wolfe algorithms

We now consider Frank-Wolfe algorithms (also known as conditional gradient algorithms) for minimizing a smooth convex function $f$ on a compact convex set $\mathcal{K}$. These algorithms are dedicated to situations where one can easily minimize linear functions on $\mathcal{K}$ (see, e.g., Jaggi, 2013, and references therein). The algorithm has the following form:

$$\begin{aligned} \bar{x}_k &\in \arg\min_{x \in \mathcal{K}} \; f(x_{k-1}) + f'(x_{k-1})^\top (x - x_{k-1}) \\ x_k &= (1 - \rho_k) x_{k-1} + \rho_k \bar{x}_k. \end{aligned}$$

That is, the first order Taylor expansion of $f$ at $x_{k-1}$ is minimized, ending up typically in an extreme point $\bar{x}_k$ or $\mathcal{K}$ and a convex combination of $x_{k-1}$ and $\bar{x}_k$ is considered. While some form of line search can be used to find $\rho_k$, we consider so-called "open loop" schemes where $\rho_k = 1/k$ or $\rho_k = 2/(k+1)$ (Dunn and Harshbarger, 1978; Jaggi, 2013).

In terms of function values, these two variants are known to converge at respective rates $O(\log(k)/k)$ and $O(1/k)$. Moreover, as illustrated in Figure 4, they are known to zig-zag towards the optimal point. Avoiding this phenomenon can be done in several ways, for examples through optimizing over all convex combinations of the $\bar{x}_i$'s for $i \leqslant k$ (Von Hohenbalken, 1977), or through so-called "away steps" (Guélat and Marcotte, 1986; Lacoste-Julien and Jaggi, 2015). In this section, we consider Richardson extrapolation and assume for simplicity that $\mathcal{K}$ is a polytope (which is a typical use case for Frank-Wolfe algorithms). Note here that we are considering asymptotic convergence rates, and even without extrapolation (but with a local strong-convexity assumption), we can beat the $O(1/k)$ rates for the step-size $\rho_k = 2/(k+1)$.[1]

**Asymptotic expansion.** In order to provide the proposition below (see Appendix C for a proof) that characterizes the zig-zagging phenomenon, we assume regularity properties similar to Section 2.1

[1]Note that: (a) the lower bound with dependence $O(1/k)$ from Canon and Cullum (1968) only applies to Frank-Wolfe algorithms with line-search, and (b) that our bounds are local and the constants have to depend on dimension so as to not contradict the lower bound from Jaggi (2013).

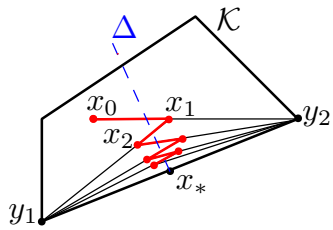


Figure 4: Frank-Wolfe algorithm zigzagging. Starting from $x_0$, the algorithm always moves towards one of the extreme points of $\mathcal{K}$, with an average direction of $\Delta$.

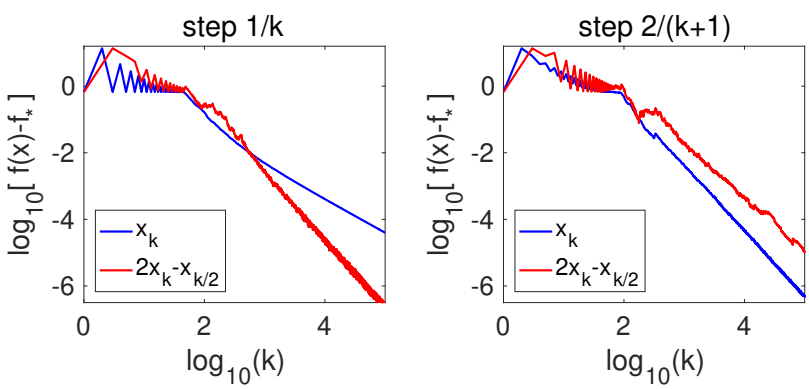


Figure 5: Frank-Wolfe for the "constrained logistic Lasso", that is, $\min_{x \in \mathbb{R}^d} f(x)$ such that $\|x\|_1 \leqslant c$, with $f(x) = \frac{1}{n} \sum_{i=1}^n \log(1 + \exp(-b_i x^\top a_i))$. We consider $n = 400$ observations in dimension $d = 400$, sampled from a standard normal distribution, and with a constraint on the $\ell_1$-norm. Left: step size $1/k$ with slopes $-1$ (blue) and $-2$ (red). Right: step size $2/(k+1)$, with slope approximately $-2$ for the two curves.

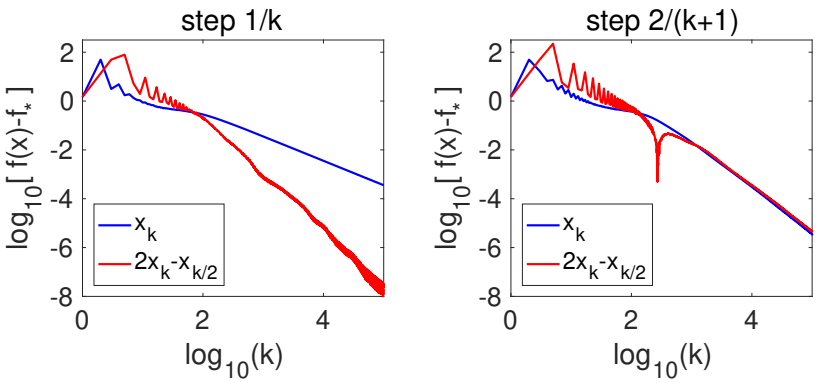


Figure 6: Frank-Wolfe for the dual of robust regression. We consider the dual of absolute loss regression with $n = 400$ observations in dimension $d = 200$, sampled from a standard normal distribution, with a squared Euclidean norm penalty. The primal problem is $\inf_{y \in \mathbb{R}^d} \frac{1}{n}\|b - Ax\|_1 + \frac{\lambda}{2}\|y\|_2^2$ while the dual problem is $\sup_{\|x\|_\infty \leqslant 1} -f(x)$, with $f(x) = -\frac{1}{n}x^\top b + \frac{1}{2n^2\lambda}x^\top AA^\top x$. Left: step size $1/k$ with slopes $-1$ (blue) and $-2$ (red). Right: step size $2/(k+1)$, with slope $-2$ for the two curves.

and that the unique minimizer is "in the middle of a face" of $\mathcal{K}$, which is often referred to as *constraint qualification* in optimization (Nocedal and Wright, 2006).

**Proposition 2** *Assume $f$ convex, three-times differentiable with bounded third-order derivatives in the polytope $\mathcal{K}$, and with a unique minimizer $x_* \in \mathbb{R}^d$ such that $f''(x_*)$ is positive definite. Moreover, we assume $x_*$ is strictly in a $(m-1)$-dimensional face of $\mathcal{K}$ which is the convex hull $\mathcal{K}_*$ of extreme points $y_1, \dots, y_m \in \mathbb{R}^d$, and for which $\min_{y \in \mathcal{K}} f'(x_*)^\top y$ is attained only by elements of $\mathcal{K}_*$. Then:*

- *For $\rho_k = 1/k$, $x_k = x_* + \frac{1}{k}\Delta_1 + O(1/k^2)$. This implies $f(x_k) - f(x_*) = \frac{1}{k}\Delta_1^\top f'(x_*) + O(1/k^2)$ and $f(2x_k - x_{k/2}) - f(x_*) = O(1/k^2)$.*
- *For $\rho_k = 2/(k+1)$, $x_k = x_* + \frac{1}{k(k+1)}\Delta_2 + O(1/k^2)$. This implies $f(x_k) - f(x_*) = O(1/k^2)$ and $f(2x_k - x_{k/2}) - f(x_*) = O(1/k^2)$.*

*The two vectors $\Delta_1$ and $\Delta_2$, are orthogonal (for the dot-product defined by $f''(x_*)$) to the span of all $y_i - x_*$, $i = 1, \dots, m$*

We now discuss the consequences of the proposition above.

**Step-size $\rho_k = 1/k$.** Although it leads to a worse performance than the step-size $\rho_k = 2/(k+1)$ both in theory (extra logarithmic term) and in practice, we consider it as (1) this is the "historical" step-size (Dunn and Harshbarger, 1978) with an interesting behavior in our set-up, and (2) the corresponding dual algorithm is the subgradient method with plain averaging (see, e.g., Bach, 2015), which is sometimes preferred in online learning (Hazan et al., 2006).

As shown in Prop. 2, Richardson extrapolation allows to go from an $O(1/k)$ to an $O(1/k^2)$ convergence rate. In the left plots of Figure 5 and Figure 6, we can observe the benefits of Richardson extrapolation on two optimization problems with the step-size $\rho_k = 1/k$. Note that: (a) asymptotically, there is provably no extra logarithmic factor like we have for the non-asymptotic convergence rate, and (b) that the Richardson extrapolated iterate may not be within $\mathcal{K}$, but is $O(1/k)$ away from it (in our simulations, we simply make the iterate feasible by rescaling).

**Step-size $\rho_k = 2/(k+1)$.** As shown in Prop. 2, we already get an performance of $O(1/k^2)$ without extrapolation (which is a new asymptotic result for Frank-Wolfe algorithm on polytopes), and Richardson extrapolation does not lead to any acceleration. In the right plots of Figure 5 and Figure 6, we indeed see no benefits (but no strong degradation).

Note that here (and for both step-sizes), higher order Richardson would not lead to further cancellation as within the span of the supporting face, we have an oscillating behavior similar to the left plot of Figure 2. Moreover, although we do not have a proof, the closed loop algorithm exhibits the same behavior as the step $\rho_k = 1/k$, both with and without extrapolation, which is consistent with the analysis of Canon and Cullum (1968). It would also be interesting to consider the benefits for Richardson extrapolation for strongly-convex sets (Garber and Hazan, 2015).

# 3 Extrapolation on the Regularization Parameter

In this section, we explore the application of Richardson extrapolation to regularization methods. In a nutshell, regularization allows to make an estimation problem more stable (less subject to variations for statistical problems) or the algorithm faster (for optimization problems). However, regularization adds a bias that needs to be removed. In this section, we apply Richardson extrapolation to the regularization parameter to reduce this bias. We consider two applications where we can provably show some benefits: (a) smoothing for non-smooth optimization in Section 3.1, and (b) ridge regression in Section 3.2.

Other applications include the classical use within integration methods, where the technique is often called Richardson-Romberg extrapolation (Gautschi, 1997), and for bias removal in constant-step-size stochastic gradient descent for sampling (Durmus et al., 2016) and optimization (Dieuleveut et al., 2019).

## 3.1 Smoothing non-smooth problems

We consider the minimization of a convex function of the form $f(x) = h(x) + g(x)$, where $h$ is smooth and $g$ is non-smooth. These optimization problems are ubiquitous in machine learning and signal processing, where the lack of smoothness can come from (a) non-smooth losses such as max-margin losses used in support vector machines and more generally structured output classification (Taskar et al., 2005; Tsochantaridis et al., 2005), and (b) sparsity-inducing regularizers (see, e.g., Bach et al., 2012, and references therein). While many algorithms can be used to deal with this non-smoothness, we consider a classical smoothing technique below.

**Nesterov smoothing.** In this section, we consider the smoothing approach of Nesterov (2005) where the non-smooth term is "smoothed" into $g_\lambda$, where $\lambda$ is a regularization parameter, and accelerated gradient descent is used to minimize $h + g_\lambda$.

A typical way of smoothing the function $g$ is to add $\lambda$ times a strongly convex regularizer to the Fenchel conjugate of $g$ (see an example below); as shown by Nesterov (2005), this leads to a function $g_\lambda$ which has a smoothness constant (defined as the maximum of the largest eigenvalues of all Hessians) proportional to $1/\lambda$, with a uniform error of $\lambda$ between $g$ and $g_\lambda$. Given that accelerated gradient descent leads to an iterate with excess function values proportional to $1/(\lambda k^2)$ after $k$ iterations, with the choice of $\lambda \propto 1/k$, this leads to an excess in function values proportional to $1/k$, which improves on the subgradient method which converges in $O(1/\sqrt{k})$.

**Richardson extrapolation.** If we denote by $x_\lambda$ the minimizer of $h + g_\lambda$ and $x_*$ the global minimizer of $f = h + g$, if we can show that $x_\lambda = x_* + \lambda\Delta + O(\lambda^2)$, then $x_\lambda^{(1)} = 2x_\lambda - x_{2\lambda} = O(\lambda^2)$ and we can expand $f(x_\lambda^{(1)}) = f(x_*) + O(\lambda^2)$, which is better than the $O(\lambda)$ approximation without extrapolation.

Then, with $\lambda \propto k^{-2/3}$, to balance the two terms $1/(\lambda k^2)$ and $\lambda^2$, we get an overall convergence rate for the non-smooth problem of $k^{-4/3}$. We now make this formal for the special (but still quite generic) case of polyhedral functions $g$, and also consider $m$-step Richardson extrapolation, which leads to a convergence rate arbitrarily close to $O(1/k^2)$.

**Polyhedral functions.** We consider a polyhedral function of the form

$$g(x) = \max_{i \in \{1,\dots,m\}} a_i^\top x - b_i = \max_{i \in \{1,\dots,m\}} (Ax - b)_i,$$

where $A \in \mathbb{R}^{m \times d}$ and $b \in \mathbb{R}^m$. This form includes traditional regularizers such as the $\ell_1$-norm, the $\ell_\infty$-norm, grouped $\ell_1$-$\ell_\infty$-norms (Negahban and Wainwright, 2008), or more general sparsity-inducing norms (Bach, 2010).

We consider the smoothing of $g$ as:

$$g_\lambda(x) = \max_{\eta \in \Delta_m} \eta^\top (Ax - b) - \lambda \varphi(\eta),$$

for some strongly convex function $\varphi$, typically, the negative entropy $\sum_{i=1}^m \{\eta_i \log \eta_i - \eta_i\}$, or $\frac{1}{2}\|\eta\|_2^2$. For our asymptotic expansion, we also need a form a constraint qualification (see proof in Appendix D).

**Proposition 3** *Assume $h$ convex, three-times differentiable with bounded third-order derivatives $g$ convex, and a unique minimizer $x_* \in \mathbb{R}^d$ of $h + g$ such that $h''(x_*)$ is positive definite. Assume there exists $\eta_* \in \mathbb{R}^m$ in the simplex $\Delta_m$ (defined as the vectors in $\mathbb{R}^m$ with non-negative components summing to one), such that, for the support $I \subset \{1, \dots, m\}$ of $\eta_*$ (that is, the set of non-zeros),*

$$h'(x_*) + A^\top \eta_* = 0,$$

*and*

$$\max_{i \in \{1,\dots,m\}} (Ax_* - b)_i \text{ is only attained for all } i \in I.$$

*Assume moreover the submatrix $A_I$ obtained by taking the the rows of $A$ indexed by $I$ has full rank. We denote by $x_\lambda$ a minimizer of $h(x) + g_\lambda(x)$, and $\eta_\lambda$ the corresponding dual variable. Then:*

$$x_\lambda = x_* + \lambda \Delta + O(\lambda^2),$$

*with $\Delta = h''(x_*)^{-1} A_I^\top \big[A_I h''(x_*)^{-1} A_I^\top\big]^{-1} (\eta_*)_I$ for the quadratic penalty, and a similar expression for the entropic penalty.*

The proposition above implies that (see detailed proof in Appendix D for details) the smoothing technique asymptotically adds a bias of order $\lambda$

$$f(x_\lambda) = f(x_*) + O(\lambda),$$

where we recover the usual upper-bound in $O(\lambda)$, confirming the result from Nesterov (2005). The key other consequence is that

$$f(x_\lambda^{(1)}) = f(2x_\lambda - x_{2\lambda}) = f(x_*) + O(\lambda^2),$$

which shows the benefits of Richardson extrapolation.

**Multiple-step Richardson extrapolation.** Given that one-step Richardson extrapolation allows to go from a bias of $O(\lambda)$ to $O(\lambda^2)$, a natural extension is to consider $m$-step Richardson extrapolation (Gautschi, 1997), that is, a combination of $m+1$ iterates:

$$x_\lambda^{(m)} = \sum_{i=1}^{m+1} \alpha_i^{(m)} x_{i\lambda},$$

where the weights $\alpha_i^{(m)}$ are such that the first $m$ powers in the Taylor expansion of $x_\lambda$ cancel out. This can be done by solving the linear system based on the following equations:

$$\sum_{i=1}^{m+1} \alpha_i^{(m)} = 1 \tag{1}$$

$$\forall j \in \{1, \dots, m\}, \qquad \sum_{i=1}^{m+1} \alpha_i^{(m)} i^j = 0. \tag{2}$$

Using the same technique as Pagès (2007, Lemma 3.1), this is a Vandermonde system with a closed form solution (see proof in Appendix E.3):

$$\alpha_i^{(m)} = (-1)^{i-1} \binom{m+1}{i}.$$

We show in Appendix D the following proposition:

**Proposition 4** *On top of assumptions from Prop. 3, assume $h$ is $(m+2)$-times differentiable with bounded derivatives. Then*

$$f(x_\lambda) = f(x_*) + O(\lambda^{m+1}).$$

Thus, within the smoothing technique, if we consider $\lambda \propto 1/k^{2/(m+2)}$, to balance the terms $1/(\lambda k^2)$ and $\lambda^{m+1}$, we get an error for the non-smooth problem of $1/k^{2(m+1)/(m+2)}$, which can get arbitrarily close to $1/k^2$ when $m$ gets large. The downsides are that (a) the constants in front of the asymptotic equivalent may blow up (a classical problem in high-order expansions), and (b) $m$-step extrapolation requires running the algorithm $m$ times (this can be down in parallel). In our experiments below, 3-step extrapolation already brings in most of the benefits.

**Experiments.** We consider the penalized Lasso problem:

$$\min_{x \in \mathbb{R}^d} \frac{1}{2n} \sum_{i=1}^n (b_i - x^\top a_i)^2 + \lambda \|x\|_1,$$

where $(a_i, b_i) \in \mathbb{R}^d \times \mathbb{R}$ for $i = 1, \dots, n$, for $d = 100$ and $n = 100$, and with input data distributed as a standard normal vector. We use either a dual entropic penalty or a dual quadratic penalty for smoothing each component $|x_j|$ of the $\ell_1$-norm of $x$. Plots for the quadratic penalty are presented here in Figure 7, while plots for the entropic penalty are presented in Figure 9 in Appendix D with the same conclusions.

In the left plot of Figure 7, we illustrate the dependence of $f(x_\lambda) - f(x_*)$ on $\lambda$ for Richardson extrapolation with various orders, while in the right plot of Figure 7, we study the effect of extrapolation to solve the non-smooth problem. For a series of regularization parameters equal to $2^i$ for $i$ between $-18$ and 1 (sampled every 1/5), we run accelerated gradient descent on $h + g_\lambda$ and we plot the value of $f(x) - f(x_*)$ for the various estimates, where for each number of iterations, we minimize over the regularization parameter. This is an oracle version of varying $\lambda$ as a function of the number of iterations (a detailed evaluation where $\lambda$ depends on $k$ could also be carried out). In Figure 7, we plot the excess function values as a function of the number of iterations, *taking into account that $m$-step Richardson extrapolation requires $m$-times more iterations*. We see that we get a strong improvement approaching $1/k^2$.

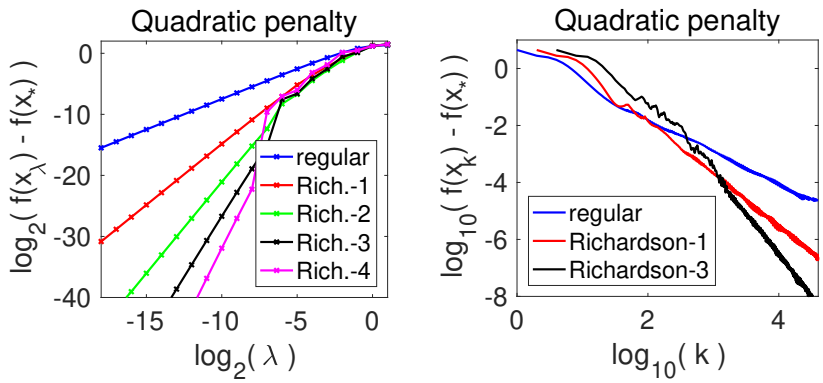


Figure 7: Richardson extrapolation for Nesterov smoothing on a penalized Lasso problem, with regularization by the quadratic penalty. Left: dependence of $f(x_\lambda) - f(x_*)$ on $\lambda$, for Richardson extrapolation of order $m$, we indeed recover a slope of $m+1$ in the log-log plot. Right: Optimization error vs. number of iterations; where we go from a slope of -1 (blue curves) to improved slopes of $-4/3$ (red curve) and $-8/5$ (black curve). See text for details.

**From non-linear programming to linear programming.** When we use the entropic penalty, the smoothing framework is generally applicable in most non-linear programming problems (see, e.g., Cominetti and Dussault, 1994). It is interesting to note that typically when applying the entropic penalty, the deviation to the global optimizer is going to zero *exponentially* in $-1/\lambda$ for some of the components (see a proof for our particular case in Appendix D), but not for the corresponding dual problem (which is our primal problem).

Another classical instance of entropic regularization in machine learning leads to the Sinkhorn algorithm for computing optimal transport plans (Cuturi, 2013). For that problem, the entropic penalty is put directly on the original problem, and the deviation in estimating the optimal transport plan can be shown to be asymptotically exponential in $-1/\lambda$ (Cominetti and San Martín, 1994), and thus there Richardson extrapolation is not helpful (unless one wants to estimate the Kantorovich dual potentials).

## 3.2 Improving bias in ridge regression

We consider the ridge regression problem, that is, $w_\lambda$ is the unique minimizer of

$$\min_{w \in \mathbb{R}^d} \frac{1}{2n} \|y - \Phi w\|_2^2 + \frac{\lambda}{2} \|w\|_2^2,$$

where $\Phi \in \mathbb{R}^{n \times d}$ is a feature vector and $y \in \mathbb{R}^n$ a vector of responses (Friedman et al., 2001). The solution may be obtained in closed form by solving the normal equations, as $w_\lambda = (\Phi^\top \Phi + n\lambda I)^{-1} \Phi^\top y$.

The regularization term $\frac{\lambda}{2}\|w\|_2^2$ is added to avoid overfitting and control the variability of $w_\lambda$ due to the randomness in the training data (the higher the $\lambda$, the more control); however, it does create a bias that goes down as $\lambda$ goes to zero. Richardson extrapolation can be used to reduce this bias. We thus consider $w_\lambda^{(1)} = 2w_\lambda - w_{2\lambda}$ and more generally

$$w_\lambda^{(m)} = \sum_{i=0}^{m} \alpha_i^{(m)} w_{i\lambda},$$

with the same weights as defined in Eq. (1) and Eq. (2). In order to compute $w_\lambda^{(m)}$, either $m$ ridge regression problems can be solved, or a closed-form spectral formula can be used based on a single singular value decomposition of the kernel matrix (see Appendix E.3 for details).

**Theoretical analysis.** Following Bach (2013), we assume for simplicity that $\Phi$ is deterministic and that $y = z + \varepsilon$, where $\varepsilon$ has zero mean and covariance matrix $\sigma^2 I$. We consider the in-sample error of $\hat{y}_\lambda = \Phi w_\lambda = K(K + n\lambda I)^{-1} y = \hat{H}_\lambda y$, where $K = \Phi\Phi^\top$ is the usual kernel matrix, and $\hat{H}_\lambda$ is the smoothing matrix, which is equal to $I$ for very small $\lambda$ and equal to zero for very large $\lambda$. We consider the so-called "in-sample generalization error", that is, we want to minimize

$$\frac{1}{n}\mathbb{E}\|\hat{y}_\lambda - z\|_2^2 \quad = \quad \mathrm{bias}(\hat{H}_\lambda) + \mathrm{variance}(\hat{H}_\lambda),$$

where

$$\begin{aligned} \mathrm{bias}(\hat{H}_\lambda) &= \frac{1}{n}\|(\hat{H}_\lambda - I)z\|_2^2 \\ \mathrm{variance}(\hat{H}_\lambda) &= \frac{\sigma^2}{n}\operatorname{tr}\hat{H}_\lambda^2. \end{aligned}$$

The bias term is increasing in $\lambda$, while the variance term in decreasing in $\lambda$, and there is thus a trade-off between these two terms. To find the optimal $\lambda$, assumptions need to be made on the problem, regarding the eigenvalues of $K$, and the components of $z$ in the eigenbasis of $K$. That is, following the notations of Bach (2013, Section 4.3), we assume that the eigenvalues of $K$ are $\Theta(n\mu_i)$ (that is bounded from above and below by constants times $n\mu_i$) and the coordinates of $z$ in the eigenbasis of $K$ are $\Theta(\sqrt{n\nu_i})$. The precise trade-off depends on the rates at which $\mu_i$ and $\nu_i$ decay to zero.

A classical situation is $\mu_i \sim i^{-2\beta}$ and $\nu_i \sim i^{-2\delta}$, where $\beta > 1/2$ and $\delta > 1/2$ (to ensure finite energy). As detailed in Appendix E:

- the variance term is equivalent to $\frac{\sigma^2}{n}\lambda^{-1/2\beta}$ and does not depend on $z$ or $\delta$;
- the bias term depends on both $\delta$ and $\beta$: for signals which are not too smooth (i.e., not too fast decay of $\nu_i$, and thus small $\delta$), that is if $\delta < 2\beta + 1/2$, then the bias term is equivalent to $\lambda^{(2\delta-1)/2\beta}$ and we can thus find the optimal $\lambda$ as $(\sigma^2/n)^{\beta/\delta}$ leading to predictive performance of $(\sigma^2/n)^{1-1/2\delta}$, which happens to be optimal (Caponnetto and De Vito, 2007). However, when $\delta > 2\beta + 1/2$, a phenomenon called "saturation" occurs, and the bias term is equivalent to $\lambda^2$ (independent of $\delta$), and the optimized predictive performance is $(\sigma^2/n)^{1-1/(4\beta+1)}$, which is not optimal anymore.

As shown in Appendix E, by reducing the bias, with $m$-step Richardson interpolation, we can show that the variance term is bounded by a constant times the usual one, while the bias term is equivalent to $\lambda^{(2\delta-1)/2\beta}$ for a wider range of $\delta$, that is, $\delta < 2(m+1)\beta + 1/2$, which recovers for $m = 0$ the non-extrapolated estimate. This leads to optimal statistical performance for a wider range of problems.

**Experiments.** As an illustration, we consider a ridge regression problem with data uniformly sampled on the unit sphere in dimension $d = 40$ with $n = 200$ observations, and $y$ generated as a linear function of the input plus some noise. We consider the rotation invariant kernel equal to the expectation $k(x, x') = \mathbb{E}_\tau(1 + x^\top x')\sigma(\tau^\top x)\sigma(\tau^\top x')$, for $\tau$ uniform on the sphere. This is equal to, for $\sigma(\alpha) = 1_{\alpha>0}$ (see Cho and Saul, 2009; Bach, 2017):

$$k(x, y) \propto (1 + x^\top x')\big[\pi - \arccos(x^\top x')\big].$$

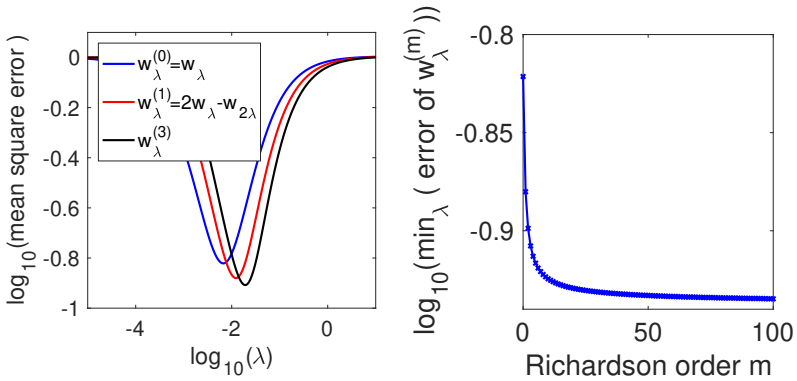


Figure 8: Left: regularization path for the classical iterate $w_\lambda$, one step of Richardson $w_\lambda^{(1)}$ and 3 steps $w_\lambda^{(3)}$. Right: optimal error as a function of the order of the Richardson step.

When the number of observations $n$ tends to infinity, the eigenvalues of $\frac{1}{n}K$ are known to converge to the eigenvalues of a certain infinite-dimensional operator (Koltchinskii and Giné, 2000). As shown by Bach (2017), the corresponding eigenvalues of the kernel matrix decay as $i^{-1-1/d}$. We consider $z$ generated as a linear function so that $\nu_i$ has a finite number of non zero components in the eigenbasis of $K$.

In the left plot of Figure 8, we consider 1-step and 3-step Richardson extrapolation and plot the generalization error (averaged over 10 replications) as a function of the regularization parameter: we can see that as expected, (a) with extrapolation the curves move to the right (we can use a larger $\lambda$ for a similar performance, which is advantageous as iterative algorithms are typically faster), and (b) the minimal error is smaller (which is true here because we learn a smooth function). In the right plot of Figure 8, we study the effect of increasing the order $m$ of extrapolation, showing that the larger the better with some saturation. With $m$ infinite, there will be overfitting as the corresponding spectral filter is non-stable, but this happens very slowly (see Appendix E.3 for details).

# 4 Conclusion

In this paper, we presented various applications of Richardson extrapolation to machine learning optimization and estimation problems, each time with an asymptotic analysis showing the potential benefits. For example, when using the number of iterations of an iterative algorithm to perform extrapolation, we can accelerate Frank-Wolfe algorithms to have an asymptotic rates of $O(1/k^2)$ on polytopes for the step-size $\rho_k = 1/k$ and locally strongly-convex functions (this is achieved without extrapolation for the step-size $\rho_k = 2/(k+1)$). When extrapolating based on the regularization parameter, we can accelerate Nesterov smoothing technique to have asymptotic rates close to $O(1/k^2)$.

We also highlighted situations where Richardson extrapolation does not bring any benefits (but does not degrade performance much), namely when applied to accelerated gradient descent or the Sinkhorn algorithm for optimal transport.

The analysis in this paper can be extended in a number of ways: (1) while the paper has focused on asymptotic analysis for simplicity, non-asymptotic analysis could be carried out to study more finely when acceleration starts, (2) we have focused on deterministic optimization algorithms, and extensions to stochastic algorithms could be derived, along the lines of the work of Dieuleveut et al. (2019), (3) we have primarily focused on convex optimization algorithms but non-convex extensions, like done by Scieur et al. (2018) for Anderson acceleration, could also lead to acceleration

## Acknowledgements

This work was funded in part by the French government under management of Agence Nationale de la Recherche as part of the "Investissements d'avenir" program, reference ANR-19-P3IA-0001 (PRAIRIE 3IA Institute). We also acknowledge support the European Research Council (grant SEQUOIA 724063).

# References

Milton Abramowitz, Irene A. Stegun, and Robert H. Romer. *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables*. Dover Publications, 1988.

Alexander Craig Aitken. On Bernoulli's numerical solution of algebraic equations. *Proceedings of the Royal Society of Edinburgh*, 46:289–305, 1927.

Donald G Anderson. Iterative procedures for nonlinear integral equations. *Journal of the ACM (JACM)*, 12(4):547–560, 1965.

Francis Bach. Structured sparsity-inducing norms through submodular functions. In *Advances in Neural Information Processing Systems*, pages 118–126, 2010.

Francis Bach. Sharp analysis of low-rank kernel matrix approximations. In *Conference on Learning Theory*, 2013.

Francis Bach. Duality between subgradient and conditional gradient methods. *SIAM Journal on Optimization*, 25(1):115–129, 2015.

Francis Bach. Breaking the curse of dimensionality with convex neural networks. *Journal of Machine Learning Research*, 18(1):629–681, 2017.

Francis Bach and Eric Moulines. Non-asymptotic analysis of stochastic approximation algorithms for machine learning. In *Advances in Neural Information Processing Systems*, 2011.

Francis Bach, Rodolphe Jenatton, Julien Mairal, and Guillaume Obozinski. Optimization with sparsity-inducing penalties. *Foundations and Trends® in Machine Learning*, 4(1):1–106, 2012.

Michael D. Canon and Clifton D. Cullum. A tight upper bound on the rate of convergence of Frank-Wolfe algorithm. *SIAM Journal on Control*, 6(4):509–516, 1968.

Andrea Caponnetto and Ernesto De Vito. Optimal rates for the regularized least-squares algorithm. *Foundations of Computational Mathematics*, 7(3):331–368, 2007.

Yutian Chen, Max Welling, and Alex Smola. Super-samples from kernel herding. In *Conference on Uncertainty in Artificial Intelligence*, 2010.

Youngmin Cho and Lawrence K. Saul. Kernel methods for deep learning. In *Advances in Neural Information Processing Systems*, 2009.

Roberto Cominetti and Jean-Pierre Dussault. Stable exponential-penalty algorithm with superlinear convergence. *Journal of Optimization Theory and Applications*, 83(2):285–309, 1994.

Roberto Cominetti and Jaime San Martín. Asymptotic analysis of the exponential penalty trajectory in linear programming. *Mathematical Programming*, 67(1-3):169–187, 1994.

Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. In *Advances in Neural Information Processing Systems*, 2013.

Aymeric Dieuleveut, Alain Durmus, and Francis Bach. Bridging the gap between constant step size stochastic gradient descent and Markov chains. *Annals of Statistics*, In Press, 2019.

Joseph C. Dunn and S. Harshbarger. Conditional gradient algorithms with open loop step size rules. *Journal of Mathematical Analysis and Applications*, 62(2):432–444, 1978.

Alain Durmus, Umut Simsekli, Eric Moulines, Roland Badeau, and Gaël Richard. Stochastic gradient Richardson-Romberg Markov chain Monte Carlo. In *Advances in Neural Information Processing Systems*, 2016.

Nicolas Flammarion and Francis Bach. From averaging to acceleration, there is only a step-size. In *Conference on Learning Theory*, 2015.

Jerome Friedman, Trevor Hastie, and Robert Tibshirani. *The Elements of Statistical Learning*. Springer series in statistics New York, 2001.

Dan Garber and Elad Hazan. Faster rates for the Frank-Wolfe method over strongly-convex sets. In *International Conference on Machine Learning*, 2015.

Walter Gautschi. *Numerical Analysis*. Springer Science & Business Media, 1997.

Gene H. Golub and Charles F. Van Loan. *Matrix Computations*. Johns Hopkins University Press, 1989.

Jacques Guélat and Patrice Marcotte. Some comments on Wolfe's away step. *Mathematical Programming*, 35(1):110–119, 1986.

Elad Hazan, Adam Kalai, Satyen Kale, and Amit Agarwal. Logarithmic regret algorithms for online convex optimization. In *Conference on Computational Learning Theory*, pages 499–513, 2006.

Martin Jaggi. Revisiting Frank-Wolfe: Projection-free sparse convex optimization. In *International Conference on Machine Learning*, 2013.

Prateek Jain, Sham M. Kakade, Rahul Kidambi, Praneeth Netrapalli, and Aaron Sidford. Parallelizing stochastic gradient descent for least squares regression: Mini-batching, averaging, and model misspecification. *Journal of Machine Learning Research*, 18(223):1–42, 2018.

D. C. Joyce. Survey of extrapolation processes in numerical analysis. *Siam Review*, 13(4):435–490, 1971.

Vladimir Koltchinskii and Evarist Giné. Random matrix approximation of spectra of integral operators. *Bernoulli*, 6(1):113–167, 2000.

Simon Lacoste-Julien and Martin Jaggi. On the global linear convergence of Frank-Wolfe optimization variants. In *Advances in Neural Information Processing Systems*, 2015.

Sahand Negahban and Martin J. Wainwright. Joint support recovery under high-dimensional scaling: Benefits and perils of $\ell_{1,\infty}$-regularization. In *Advances in Neural Information Processing Systems*, pages 1161–1168, 2008.

Arkadi Nemirovski, Anatoli Juditsky, Guanghui Lan, and Alexander Shapiro. Robust stochastic approximation approach to stochastic programming. *SIAM Journal on Optimization*, 19(4):1574–1609, 2009.

Yu Nesterov. Smooth minimization of non-smooth functions. *Mathematical Programming*, 103(1): 127–152, 2005.

Yurii Nesterov. *Introductory Lectures on Convex Optimization: A Basic Course*, volume 87. Springer Science & Business Media, 2013.

Yurii E. Nesterov. A method for solving the convex programming problem with convergence rate $O(1/k^2)$. In *Doklady Akademii Nauk SSSR*, volume 269, pages 543–547, 1983.

Jorge Nocedal and Stephen Wright. *Numerical Optimization*. Springer Science & Business Media, 2006.

Gilles Pagès. Multi-step Richardson-Romberg extrapolation: remarks on variance control and complexity. *Monte Carlo Methods and Applications*, 13(1):37–70, 2007.

Boris T. Polyak and Anatoli B. Juditsky. Acceleration of stochastic approximation by averaging. *SIAM Journal on Control and Optimization*, 30(4):838–855, 1992.

Lewis Fry Richardson. The approximate arithmetical solution by finite differences of physical problems involving differential equations, with an application to the stresses in a masonry dam. *Philosophical Transactions of the Royal Society of London. Series A*, 210(459-470):307–357, 1911.

D. Ruppert. Efficient estimations from a slowly convergent Robbins-Monro process. Technical Report 781, Cornell University Operations Research and Industrial Engineering, 1988.

Damien Scieur, Alexandre d'Aspremont, and Francis Bach. Regularized nonlinear acceleration. In *Advances In Neural Information Processing Systems*, 2016.

Damien Scieur, Edouard Oyallon, Alexandre d'Aspremont, and Francis Bach. Nonlinear acceleration of deep neural networks. Technical Report 1805.09639, arXiv, 2018.

Weijie Su, Stephen Boyd, and Emmanuel J. Candès. A differential equation for modeling Nesterov's accelerated gradient method: theory and insights. *Journal of Machine Learning Research*, 17(1): 5312–5354, 2016.

Ben Taskar, Vassil Chatalbashev, Daphne Koller, and Carlos Guestrin. Learning structured prediction models: A large margin approach. In *International Conference on Machine Learning*, 2005.

Ioannis Tsochantaridis, Thorsten Joachims, Thomas Hofmann, and Yasemin Altun. Large margin methods for structured and interdependent output variables. *Journal of Machine Learning Research*, 6(Sep):1453–1484, 2005.

Balder Von Hohenbalken. Simplicial decomposition in nonlinear programming algorithms. *Mathematical Programming*, 13(1):49–68, 1977.

# A Preliminary considerations

We first start with lemmas that we will need in subsequent proofs; the second one shows strong convexity on a level set once we assume that the Hessian at optimum is positive definite.

**Lemma 1** *Assume $f$ convex, three-times differentiable with bounded third-order derivatives, and a point $x_*$ such that $f''(x_*)$ is positive definite. Then there exists $c > 0$ such that for any $x \in \mathbb{R}^d$.*

$$\frac{1}{2}(x - x_*)^\top f''(x_*)(x - x_*) \leqslant c \Rightarrow f''(x) \succcurlyeq \frac{1}{2} f''(x_*).$$

**Proof** Since $f''(x_*)$ is positive definite, $\lambda_{\min}(f''(x_*)) > 0$, and $\frac{1}{2}(x - x_*)^\top f''(x_*)(x - x_*) \leqslant c$ implies that $\|x - x_*\|_2^2 \leqslant \frac{2c}{\lambda_{\min}(f''(x_*))}$. Thus, since third order derivatives of $f$ are bounded, if $\frac{1}{2}(x - x_*)^\top f''(x_*)(x - x_*) \leqslant c$, we have $\|f''(x) - f''(x_*)\|_{\text{op}} \leqslant Ac$ for some constant $A$, and thus

$$f''(x) \succcurlyeq f''(x_*) - AcI \succcurlyeq f''(x_*) - \frac{Ac}{\lambda_{\min}(f''(x_*))} f''(x_*) \succcurlyeq \frac{1}{2} f''(x_*)$$

if $Ac \leqslant \frac{1}{2}\lambda_{\min}(f''(x_*))$. ■

**Lemma 2** *Assume $f$ convex, three-times differentiable with bounded third-order derivatives, and that $x_* \in \mathbb{R}^d$ is a minimizer of $f$ on $\mathbb{R}^d$, such that $f''(x_*)$ is positive definite. Then $x_*$ is the unique minimizer of $f$ and there exists $c > 0$ such that for any $x \in \mathbb{R}^d$,*

$$f(x) - f(x_*) \leqslant c \Rightarrow f''(x) \succcurlyeq \frac{1}{2} f''(x_*).$$

**Proof** Using the above Lemma 1, then there exists $c > 0$ such that

$$\frac{1}{2}(x - x_*)^\top f''(x_*)(x - x_*) \leqslant c \Rightarrow f''(x) \succcurlyeq \frac{1}{2} f''(x_*).$$

We now show that $f(x) - f(x_*) \leqslant c/2$ implies that we stay in the region where $\frac{1}{2}(x - x_*)^\top f''(x_*)(x - x_*) \leqslant c$. Indeed, using the Taylor formula with integral remainder, for $\frac{1}{2}(x - x_*)^\top f''(x_*)(x - x_*) \leqslant c$ (where the lower bound on Hessians above holds), we have

$$\begin{aligned} f(x) - f(x_*) &= 0 + \int_0^1 (x - x_*)^\top f''(x_* + t(x - x_*))(x - x_*)(1 - t)dt \\ &\geqslant \frac{1}{2}\Big(\int_0^1 (1 - t)dt\Big)(x - x_*)^\top f''(x_*)(x - x_*) = \frac{1}{4}(x - x_*)^\top f''(x_*)(x - x_*) \end{aligned}$$

and thus $f(x) - f(x_*) \leqslant c/2$ implies $\frac{1}{2}(x - x_*)^\top f''(x_*)(x - x_*) \leqslant c$, and we get the desired result. ■

# B Proof of Prop. 1 (averaged gradient descent)

In this particular case of unconstrained gradient descent, $f(x_k) - f(x_*) \leqslant \frac{1}{\gamma k}\|x_0 - x_*\|^2$, as soon as $\gamma \leqslant 1/L$ (Nesterov, 2013). This implies from Lemma 2 in Appendix A, that for $k$ larger than some $k_0$, all iterates are such that $f''(x_k) \succcurlyeq \frac{1}{2} f''(x_*)$, and thus, after that $k$, we are in the strongly convex case, where $\|x_k - x_*\| \leqslant c'\rho^k$, where $c', \rho$ depend on the lowest eigenvalue $\mu$ of $f''(x_*)$ as $c' = \|x_{k_0} - x_*\|\rho^{-k_0}$ and $\rho = (1 - \gamma\mu/2)$.

Thus, with $\bar{x}_k = \frac{1}{k}\sum_{i=0}^{k-1} x_i$, $k(\bar{x}_k - x_*)$ tends to $\sum_{i=0}^{\infty}(x_i - x_*)$ when $k \to +\infty$ (since the series is convergent), and

$$k(\bar{x}_k - x_*) - \sum_{i=0}^{\infty}(x_i - x_*) = -\sum_{i=k}^{\infty}(x_i - x_*),$$

leading to $k(\bar{x}_k - x_*) - \sum_{i=0}^{\infty}(x_i - x_*) = O(\rho^k)$, and thus, with $\Delta = \sum_{i=0}^{\infty}(x_i - x_*)$ (which is hard to compute a priori),

$$\bar{x}_k = x_* + \frac{1}{k}\Delta + O(\rho^k).$$

# C Proof of Prop. 2 (Frank-Wolfe)

**Preliminary remarks.** We consider the step-sizes $\rho_k = \frac{1}{k}$ and $\rho_k = \frac{2}{k+1}$, for which the respective convergence rates for $f(x_k) - f(x_*)$ are of the form $\frac{c}{k}$ and $c\frac{\log k}{k}$, for constants $c$ depending on the smoothness of $f$ and the diameter of the compact set (see, e.g., Jaggi, 2013). When running Frank-Wolfe (with any of the classical versions with open-loop step-sizes), we thus have $f(x_k) - f(x_*) = O((\log k)^\beta/k)$ with $\beta \in \{0, 1\}$.

Because of affine invariance of the Frank-Wolfe algorithm (and because $f''(x_*)$ is invertible), we can assume without loss of generality that $f''(x_*) = I$. Moreover, the constraint qualification implies that $f'(x_*) \neq 0$, thus, using Taylor expansion with integral remainder like in Lemma 2, if $\|x - x_*\|^2$ is small enough then

$$f(x) - f(x_*) \geqslant f'(x_*)^\top(x - x_*) + \frac{1}{4}\|x - x_*\|_2^2.$$

Since $x_*$ is the minimizer of $f$ on $\mathcal{K}$ and $x \in \mathcal{K}$, then $f'(x_*)^\top(x - x_*) \geqslant 0$ and thus, we have that $x_k - x_* = O((\log k)^{\beta/2}/\sqrt{k})$ and (from Lemma 1), that $f$ is locally strongly-convex.

**Analysis of Frank-Wolfe.** The assumption which is made implies that there exists $\sigma > 0$ such that the ball of center $x_*$ and radius $\sigma$ intersected with the affine hull of $y_1, \dots, y_m$ is included in the convex hull of $y_1, \dots, y_m$, as well $\alpha \in (0, 1)$ such that if $\cos(f'(x_*), z) \geqslant 1 - \alpha$, then $\min_{y \in \mathcal{K}} z^\top y$ is attained only by elements of the convex hull of $y_1, \dots, y_m$.

Thus, for $k$ large enough, that is greater than some $k_0$ (which can be quantified from $\sigma$, $\alpha$ and other quantities), all elements of $\arg\min_{y \in \mathcal{K}} f'(x_{k-1})^\top y$ are in the convex hull of $y_1, \dots, y_m$, that is, only the correct extreme points are selected. Denoting $\Pi$ the orthogonal projection on the span of $y_1 - x_*, \dots, y_m - x_*$, we have:

$$x_k = (1 - \rho_k)x_{k-1} + \rho_k \bar{x}_k \text{ where } \bar{x}_k \in \arg\min_{y \in \mathcal{K}} f'(x_{k-1})^\top y,$$

and thus, subtracting $x_*$:

$$x_k - x_* = (1 - \rho_k)(x_{k-1} - x_*) + \rho_k(\bar{x}_k - x_*),$$

leading to, using the projections $\Pi$ and $I - \Pi$:

$$\Pi(x_k - x_*) = (1 - \rho_k)\Pi(x_{k-1} - x_*) + \rho_k(\bar{x}_k - x_*),$$

and, because $\bar{x}_k - x_*$ is in the span of $y_1 - x_*, \dots, y_m - x_*$,

$$(I - \Pi)(x_k - x_*) = (1 - \rho_k)(I - \Pi)(x_{k-1} - x_*).$$

We now consider these two terms separately.

**Convergence of $(I-\Pi)(x_k - x_*)$.** For $\rho_k = \frac{2}{k+1}$, we have, in closed form for $k \geqslant k_0$:

$$(I-\Pi)(x_k - x_*) = \frac{k-1}{k+1}(I-\Pi)(x_{k-1} - x_*) = \frac{k_0(k_0+1)}{k(k+1)}(I-\Pi)(x_{k_0} - x_*).$$

For $\rho_k = \frac{1}{k}$, we have:

$$(I-\Pi)(x_k - x_*) = \frac{k-1}{k}(I-\Pi)(x_{k-1} - x_*) = \frac{k_0}{k}(I-\Pi)(x_{k_0} - x_*).$$

**Convergence of $\Pi(x_k - x_*)$.** We now look at the convergence of $\Pi(x_k - x_*)$. We have

$$\begin{aligned}\|\Pi(x_k - x_*)\|^2 &= \|(1-\rho_k)\Pi(x_{k-1} - x_*) + \rho_k(\bar{x}_k - x_*)\|^2 \\ &= (1-\rho_k)^2\|\Pi(x_{k-1} - x_*)\|^2 + \rho_k^2\|\bar{x}_k - x_*\|^2 + 2(1-\rho_k)\rho_k(\bar{x}_k - x_*)^\top \Pi(x_{k-1} - x_*).\end{aligned}$$

Because of the ball assumption (that is the existence of a ball around $x_*$ that is contained in the supporting face of $\mathcal{K}$), we have

$$f'(x_{k-1})^\top(x_{k-1} - \bar{x}_k) = \max_{y \in \mathcal{K}} f'(x_{k-1})^\top(x_{k-1} - y) \geqslant f'(x_{k-1})^\top(x_{k-1} - x_*) + \|\Pi f'(x_{k-1})\|\sigma,$$

which leads to

$$f'(x_{k-1})^\top(x_* - \bar{x}_k) \geqslant \|\Pi f'(x_{k-1})\|\sigma.$$

Moreover, using a Taylor expansion with $f''(x_*) = I$, we have $f'(x_{k-1}) = f'(x_*) + f''(x_*)(x_{k-1} - x_*) + O((\log k)^\beta/k) = f'(x_*) + (x_{k-1} - x_*) + O((\log k)^\beta/k)$. Moreover, by optimality of $x_*$, $f'(x_*)^\top(\bar{x}_k - x_*) = 0$. Therefore we have, for $\rho_k = O(1/k)$:

$$\begin{aligned}\|\Pi x_k - x_*\|^2 &\leqslant (1-\rho_k)^2\|\Pi x_{k-1} - x_*\|^2 + \rho_k^2 \mathrm{diam}(\mathcal{K})^2 - 2(1-\rho_k)\rho_k\|\Pi f'(x_{k-1})\|\sigma + O((\log k)^\beta/k^2) \\ &\leqslant (1-\rho_k)^2\|\Pi x_{k-1} - x_*\|^2 - 2(1-\rho_k)\rho_k\|\Pi x_{k-1} - x_*\|\sigma + O((\log k)^\beta/k^2).\end{aligned}$$

For $\rho_k = 1/k$, we get

$$k^2\|\Pi x_k - x_*\|^2 \leqslant (k-1)^2\|\Pi x_{k-1} - x_*\|^2 - 2\sqrt{(k-1)\|\Pi x_{k-1} - x_*\|^2}\sigma + O(\log k).$$

For $\rho_k = 2/(k+1)$, we get

$$k^2\|\Pi x_k - x_*\|^2 \leqslant (k-1)^2\|\Pi x_{k-1} - x_*\|^2 - 2\sqrt{(k-1)\|\Pi x_{k-1} - x_*\|^2}\sigma + O(1).$$

We then can use the following simple lemma on sequences[2]: if $u_k \geqslant 0$ such that $u_0 = 0$ and $u_k \leqslant u_{k-1} - A\sqrt{u_{k-1}} + Bv_{k-1}$ for $(v_k)$ non-decreasing and positive, then $u_k \leqslant \frac{B^2}{A^2}v_k^2 + Bv_k$.

This leads to a bound in $O(1)$ for $k^2\|\Pi x_k - x_*\|^2$ for $\rho_k = 2/(k+1)$, and in $O((\log k)^2)$ for $\rho_k = 1/k$.

Note that this is similar to the proof of the convergence of Frank-Wolfe algorithms to an interior point of the feasible set, for which we have a rate of convergence of $f(x_k) - f(x_*) = O(1/k^2)$ (Chen et al., 2010).

[2]The proof is by induction. This is true for $k = 0$. If this is true for $k-1$, then either (a) $u_{k-1} \geqslant \frac{B^2}{A^2}v_{k-1}$, then $u_k \leqslant u_{k-1} \leqslant \frac{B^2}{A^2}v_k^2 + Bv_k$ because $v_{k-1} \leqslant v_k$, or (b) $u_{k-1} \leqslant \frac{B^2}{A^2}v_{k-1}$ and then $u_k \leqslant u_{k-1} + Bv_{k-1} \leqslant \frac{B^2}{A^2}v_{k-1}^2 + Bv_{k-1} \leqslant \frac{B^2}{A^2}v_k^2 + Bv_k$.

**Putting things together.** We then have for $\rho_k = \frac{1}{k}$,

$$x_k = x_* + \frac{k_0}{k}(I - \Pi)x_{k_0} + O((\log k)^2/k^2),$$

which leads to $f(x_k) - f(x_*) = \frac{k_0}{k} f'(x_*)^\top (I - \Pi)x_{k_0} + O((\log k)^2/k^2)$, which can be put back into the original bound, to obtain the bound without the logarithmic factor (since we have now replaced $O(\log(k)/k)$ by $O((\log k)^2/k^2) = o(1/k)$ in the start of the proof). The dependence in $k$ can here lead to an acceleration.

For $\rho_k = \frac{2}{k+1}$,

$$x_k = x_* + \frac{k_0(k_0+1)}{k(k+1)}(I - \Pi)x_{k_0} + O(1/k^2),$$

which leads to $f(x_k) - f(x_*) = \frac{k_0(k_0+1)}{k(k+1)} f'(x_*)^\top (I - \Pi)x_{k_0} + O(1/k^2)$. The dependence in $k$ does not lead to an acceleration.

Note that the terms in $O(1/k^2)$ are not amenable to Richardson extrapolation because they are oscillating.

# D Proofs of Prop. 3 and Prop. 4 (Nesterov smoothing)

We denote by $x_\lambda$ the minimizer of $h(x) + g_\lambda(x)$, and $\eta_\lambda$ the corresponding dual variable. The dual variable $\eta_\lambda$ is unique because of the strong-convexity of $\varphi$, while the primal variable is unique due to the same reasoning as in Appendix A (when $\lambda$ tends to zero, $x_\lambda$ has to be close to $x_*$, and in the neighborhood of $x_*$, $h$ is strongly-convex).

The primal problem is

$$\min_{x \in \mathbb{R}^d} h(x) + \lambda \varphi^*\Big(\frac{Ax - b}{\lambda}\Big),$$

while the dual problem is

$$\max_{\eta \in \Delta_m} -\lambda\varphi(\eta) - h^*(-A^\top \eta) - \eta^\top b.$$

The primal and dual solutions $x_\lambda$ and $\eta_\lambda$ are related through duality for $\varphi$, that is,

$$\eta_\lambda = \partial\varphi^*\Big(\frac{Ax_\lambda - b}{\lambda}\Big),$$

and through duality for $h$, that is,

$$x_\lambda = \partial h^*(-A^\top \eta_\lambda).$$

Since we consider functions $\varphi$ which are uniformly bounded, then we know that $f(x_\lambda) - f(x_*) = O(\lambda)$ and thus because the function is locally strongly convex, we have: $x_\lambda - x_* = O(\lambda^{1/2})$.

## D.1 Quadratic penalty

With a quadratic penalty, and small enough $\lambda$, the solution $\eta_\lambda$ will have the same sparsity pattern, using standard techniques from active set methods (Nocedal and Wright, 2006)—that is, show that the dual solution constrained to the same active set leads to a globally optimal primal/dual solution.

Moreover, because, once restricted to $I$, the dual function is locally strongly convex, and because $\varphi$ is bounded, the deviation in dual function values is less than $O(\lambda)$ and thus $\eta_\lambda - \eta_* = O(\lambda^{1/2})$ (which is a bound we are going to improve below).

The optimality conditions for the dual problem become (stationarity with respect to $\eta_I$):

$$0 = -\lambda(\eta_\lambda)_I + A_I \partial h^*(-A_I^\top (\eta_\lambda)_I) - b_I.$$

Since $h$ is twice differentiable at $x_*$, $h''(x_*)$ is invertible, and $x_* \in \partial h^*(-A^\top \eta_*)$, by the implicit function theorem, $h^*$ is twice differentiable at $-A^\top \eta_*$, and its Hessian is $h''(x_*)^{-1}$. We can thus further expand the optimality condition above as

$$0 = -\lambda(\eta_\lambda)_I + A_I \partial h^*(-A_I^\top(\eta_*)_I) - A_I \partial^2 h^*(-A_I^\top(\eta_*)_I) A_I^\top((\eta_\lambda)_I - (\eta_*)_I) - b_I + O(\|\eta_\lambda - \eta_*\|^2).$$

Since $\partial^2 h^*(-A_I^\top(\eta_*)_I) = h''(x_*)^{-1}$ and $b_I = A_I \partial h^*(-A_I^\top(\eta_*)_I)$ (because of optimality of $x_*$ and $\eta_*$), this leads to $0 = -\lambda(\eta_\lambda)_I + A_I h''(x_*)^{-1} A_I^\top((\eta_\lambda)_I - (\eta_*)_I) + O(\|\eta_\lambda - \eta_*\|^2)$. Since we know already that $\eta_\lambda - \eta_* = O(\lambda^{1/2})$, this leads to $\eta_\lambda - \eta_* = O(\lambda)$, which in turn leads to

$$(\eta_\lambda)_I = (\eta_*)_I - \lambda\big[A_I h''(x_*)^{-1} A_I^\top\big]^{-1}(\eta_*)_I + O(\lambda^2),$$

which is the desired expansion for the dual variable. We then get:

$$x_\lambda = \partial h^*(-A^\top \eta_\lambda) = x_* + \lambda h''(x_*)^{-1} A_I^\top \big[A_I h''(x_*)^{-1} A_I^\top\big]^{-1}(\eta_*)_I + O(\lambda^2) = x_* + \lambda\Delta + O(\lambda^2).$$

Thus, $2x_\lambda - x_{2\lambda} = O(\lambda^2)$, and

$$\begin{aligned} h(x_\lambda) + g(x_\lambda) &= h(x_*) + g(x_*) + \lambda h'(x_*)^\top \Delta + g(x_* + \lambda\Delta) - g(x_*) + O(\lambda^2) \\ &= h(x_*) + g(x_*) + \big[g(x_* + \lambda\Delta) - g(x_*) + \lambda h'(x_*)^\top \Delta\big] + O(\lambda^2). \end{aligned}$$

Since by optimality $h'(x_*) \in -\partial g(x_*)$, the term $\big[g(x_* + \lambda\Delta) - g(x_*) + \lambda h'(x_*)^\top \Delta\big]$ resembles a Taylor expansion of $g$ at $x_*$, but in general, we cannot have a term in $O(\lambda^2)$ because of the non-smoothness of $g$. For example, for the $\ell_1$-norm, we get $\big[g(x_* + \lambda\Delta) - g(x_*) + \lambda f'(x_*)^\top \Delta\big] = \lambda\|\Delta_{I^c}\|_1 + \lambda f'(x_*)^\top \Delta$, is not zero, and only $O(\lambda)$.

For Richardson extrapolation, we get:

$$h(2x_\lambda - x_{2\lambda}) + g(2x_\lambda - x_{2\lambda}) \quad = \quad h(x_*) + g(x_*) + O(\lambda^2),$$

and thus an improvement from $O(\lambda)$ to $O(\lambda^2)$.

## D.2 Entropic penalty

For $\varphi(\eta) = \sum_{i=1}^m \big\{\eta_i \log \eta_i - \eta_i\big\}$, we have $\varphi'(\eta)_i = \log \eta_i$, and we cannot use anymore thet fact that $\eta_\lambda$ has the same sparsity pattern as $\eta_*$, since all components of $\eta_\lambda$ are non zero. However, since the entropy is bounded over the simplex, we still have $\eta_\lambda - \eta_* = O(\lambda^{1/2})$, and from the same reasoning as for the quadratic penalty, $x_\lambda - x_* = O(\lambda^{1/2})$.

Thus, by writing primal-dual optimality conditions, we get:

$$\begin{aligned} &-\lambda \log \eta_\lambda + A x_\lambda - b = 0 \\ &h'(x_\lambda) + A^\top \eta_\lambda = 0. \end{aligned}$$

This implies that for $i \notin I$,

$$\log \eta_i \sim (Ax_* - b)_i - \sup_j (Ax_* - b_j),$$

which is strictly negative by assumption. Thus $(\eta_{I^c})_\lambda = O(\rho^{-1/\lambda})$ for a certain $\rho \in (0,1)$. We now show that $(\eta_I)_\lambda = (\eta_*)_I + \lambda\Delta + O(\lambda^2)$. From the optimality conditions, we get:

$$0 = -\lambda(\log(\eta)_\lambda)_I + A_I \partial h^*(-A^\top \eta_\lambda) - b_I,$$

which leads to, with an additional Taylor expansion,

$$0 = -\lambda(\log(\eta)_*)_I + A_I \partial h^*(-A_I^\top(\eta_*)_I) - A_I \partial^2 h^*(-A_I^\top(\eta_*)_I) A^\top(\eta_\lambda - \eta_*) - b_I + O(\|\eta_\lambda - \eta_*\|^2) + O(\lambda\|\eta_\lambda - \eta_*\|).$$

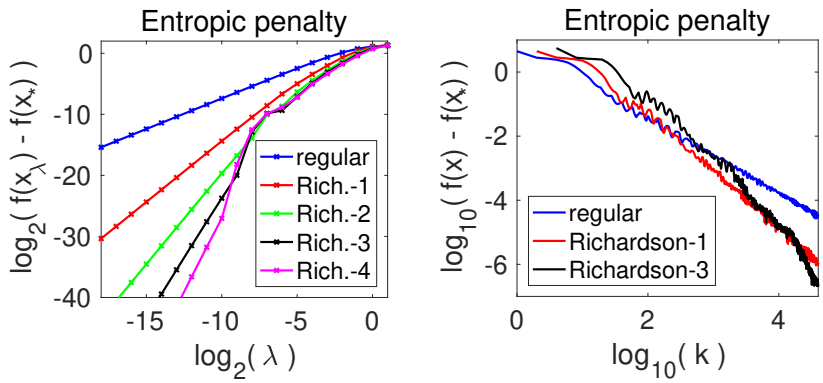


Figure 9: Richardson extrapolation for Nesterov smoothing on a penalized Lasso problem, with regularization by the entropic penalty. Left: dependence of $f(x_\lambda) - f(x_*)$ on $\lambda$, for Richardson extrapolation of order $m$, we indeed recover a slope of $m+1$ in the log-log plot. Right: Optimization error vs. number of iterations; where we go from a slope of -1 (blue curves) to improved slopes of $-4/3$ (red curve) and $-8/5$ (black curve). See text for details.

Since $\partial^2 h^*(-A_I^\top(\eta_*)_I) = h''(x_*)^{-1}$ and $b_I = A_I \partial h^*(-A_I^\top(\eta_*)_I)$ (because of optimality of $x_*$ and $\eta_*$), with moreover $(\eta_{I^c})_\lambda = O(\rho^{-1/\lambda})$, this leads to

$$-\lambda(\log(\eta)_*)_I - A_I h''(x_*)^{-1} A_I^\top((\eta_\lambda)_I - (\eta_*)_I) = O(\|(\eta_\lambda)_I - (\eta_*)_I\|^2) + O(\lambda\|(\eta_\lambda)_I - (\eta_*)_I\|) + O(\rho^{-1/\lambda}),$$

which in turn leads to

$$(\eta_\lambda)_I = (\eta_*)_I - \lambda\big[A_I f''(x_*)^{-1} A_I^\top\big]^{-1}(\log(\eta_*))_I + O(\lambda^2),$$

which is the desired expansion for the dual variable. We then get:

$$x_\lambda = \partial h^*(-A^\top \eta_\lambda) = x_* + \lambda h''(x_*)^{-1} A_I^\top \big[A_I h''(x_*)^{-1} A_I^\top\big]^{-1}(\log(\eta_*))_I + O(\lambda^2) = x_* + \lambda\Delta + O(\lambda^2).$$

Note that following Cominetti and Dussault (1994), we could get a single proof for all $\varphi(\eta) = \sum_{i=1}^n \psi(\eta_i)$, with replacing $\log \eta_*$ by $\psi'(\eta_*)$, with a condition to ensure that the zero variables lead to vanishing terms.

## D.3 Higher-order expansions (Prop. 4)

In order to use $m$-step Richardson extrapolation, we need to have a bound of the form

$$x_\lambda = x_* + \sum_{i=1}^m \Delta_i \lambda_i + O(\lambda^{m+1}).$$

We only consider for simplicity the quadratic penalty, where we simply need an expansion of $(\eta_\lambda)_I$ (minor modifications would lead to a proof for the entropic penalty since there $\eta_{I^c}$ is exponentially small).

The expansion can be obtained from the implicit function theorem applied to the equation $0 = -\lambda(\eta_\lambda)_I + A_I \partial h^*(-A_I^\top(\eta_\lambda)_I) - b_I$, which is of the form $H((\eta_\lambda)_I, \lambda) = 0$, where $H$ has high-order derivatives as long as $h^*$ is sufficiently differentiable, and the partial derivative with respect to $(\eta_\lambda)_I$ is an invertible matrix. The high-order differentiability of $h^*$ around $-A^\top \eta_*$ comes from the implicit function theorem, applied to the definition of the gradient of the Fenchel conjugate.

# E Ridge regression

## E.1 Standard extrapolation

We have $\hat{y}_\lambda = K(K+n\lambda I)^{-1}y = \hat{H}_\lambda y$, and thus $2\hat{y}_\lambda - \hat{y}_{2\lambda} = [(2\hat{H}_\lambda - \hat{H}_{2\lambda})]y$, and we can compute explicitly

$$\begin{aligned} 2\hat{H}_\lambda - \hat{H}_{2\lambda} - I &= 2K(K+n\lambda I)^{-1} - K(K+2n\lambda I)^{-1} - I \\ &= 2n\lambda\big[(K+2n\lambda I)^{-1} - (K+n\lambda I)^{-1}\big] = -2(n\lambda)^2(K+n\lambda I)^{-1}(K+2n\lambda I)^{-1}. \end{aligned}$$

Thus,

$$\begin{aligned} \mathrm{bias}(2\hat{H}_\lambda - \hat{H}_{2\lambda}) &\leqslant 4n^3\lambda^4 z^\top (K+n\lambda I)^{-4} z \\ \mathrm{variance}(2\hat{H}_\lambda - \hat{H}_{2\lambda}) &\leqslant \frac{\sigma^2}{n}\big[4\,\mathrm{tr}[K(K+n\lambda I)^{-1}]^2 + \mathrm{tr}[K(K+2n\lambda I)^{-1}]^2\big] \leqslant \frac{5\sigma^2}{n}\,\mathrm{tr}[K(K+n\lambda I)^{-1}]^2. \end{aligned}$$

We have $\mathrm{bias}(2\hat{H}_\lambda - \hat{H}_{2\lambda}) \leqslant 4\mathrm{bias}(\hat{H}_\lambda)$ and $\mathrm{variance}(2\hat{H}_\lambda - \hat{H}_{2\lambda}) \leqslant 5\mathrm{variance}(\hat{H}_\lambda)$, so the two terms never incur more than a constant factor.

However, the bias can be much improved. Following Bach (2013, Section 4.3), if the eigenvalues of $K$ are $\Theta(n\mu_i)$ and the coordinates of $z$ in the eigenbasis of $K$ are $\Theta(\sqrt{n\nu_i})$, we can compute equivalents (up to constant terms) of the bias and variance terms, for different types of decays. See Table 1 with $m=0$. Since the optimal predictive performance is $(\frac{\sigma^2}{n})^{1-1/(2\delta)}$, the only potential gains to go from $m=0$ (no extrapolation) to $m>0$ (extrapolation) occur when $(\nu_i)$ has a fast decay (that is, first, third and fifth lines).

For the first line in Table 1, we will show that the bias term for Richardson extrapolation is of the order $\lambda^4$ if $2\delta > 8\beta+1$, and equal to $\lambda^{(2\delta-1)/2\beta}$ when $2\delta < 8\beta+1$. We will thus increase the regime of validity of the bias term that leads to optimal performance, from $2\delta<4\beta+1$ to $2\delta<8\beta+1$. More generally, as we show below in Appendix E.2, the bias for $m$-step extrapolation is proportional to $n^{2m+1}\lambda^{2m+2}z^\top(K+n\lambda I)^{-2m-2}z$, and we bound it directly following closely the computations of Bach (2013, Appendix C.2):

$$\begin{aligned} n^{2m+1}\lambda^{2m+2}z^\top(K+n\lambda I)^{-2m-2}z &= n^{2m+2}\lambda^{2m+2}\sum_{i=1}^n \frac{\nu_i}{(n\mu_i+n\lambda)^{2m+2}} = \lambda^{2m+2}\sum_{i=1}^n \frac{\nu_i}{(\mu_i+\lambda)^{2m+2}} \\ &= \lambda^{2m+2}\sum_{i=1}^n \frac{i^{-2\delta}}{(i^{-2\beta}+\lambda)^4} \leqslant 2\lambda^{2m+2}\int_1^n \frac{t^{-2\delta}}{(t^{-2\beta}+\lambda)^{2m+2}}dt \\ &= 2\lambda^{2m+2}\int_1^n \frac{t^{4(m+1)\beta-2\delta}}{(1+\lambda t^{2\beta})^{2m+2}}dt. \end{aligned}$$

If $2\delta - 4(m+1)\beta > 1$, then we have an upper bound of $2\lambda^{2m+2}\displaystyle\int_1^n t^{4(m+1)\beta-2\delta}dt = O(\lambda^{2m+2})$.

If $2\delta - 4(m+1)\beta < 1$, then we can further bound

$$\begin{aligned} 2\lambda^{2m+2}\int_1^n \frac{t^{4(m+1)\beta-2\delta}}{(1+\lambda t^{2\beta})^{2m+2}}dt &= 2\lambda^{2m+2}\int_\lambda^{\lambda n^{2\beta}} \frac{[(u/\lambda)^{1/2\beta}]^{4(m+1)\beta-2\delta+1}}{(1+u)^{2m+2}}\frac{1}{2\beta}du \\ &\qquad \text{with the change of variable } u=\lambda t^{2\beta}, \\ &= 2\lambda^{2m+2-(2m+2)+\delta/\beta-1/(2\beta)}\int_\lambda^{\lambda n^{2\beta}} \frac{u^{(4(m+1)\beta-2\delta+1))/((4(m+1)\beta)}}{(1+u)^{2m+2}}\frac{1}{2\beta}du \\ &= O(\lambda^{(2\delta-1)/(2\beta)}), \end{aligned}$$

because the integral in convergent. The bias term for the third and fifth lines of Table 1 needs modifications in exactly the same way the corresponding proof from Bach (2013, Appendix C.2).

| $(\mu_i)$ | $(\nu_i)$ | var. | bias | optimal $\lambda$ | pred. perf. | condition |
|---|---|---|---|---|---|---|
| $i^{-2\beta}$ | $i^{-2\delta}$ | $\frac{\sigma^2}{n}\lambda^{-1/2\beta}$ | $\lambda^{2(m+1)}$ | $(\frac{\sigma^2}{n})^{1/(2(m+1)+1/2\beta)}$ | $(\frac{\sigma^2}{n})^{1-1/(4(m+1)\beta+1)}$ | if $2\delta > 4(m+1)\beta+1$ |
| $i^{-2\beta}$ | $i^{-2\delta}$ | $\frac{\sigma^2}{n}\lambda^{-1/2\beta}$ | $\lambda^{(2\delta-1)/2\beta}$ | $(\frac{\sigma^2}{n})^{\beta/\delta}$ | $(\frac{\sigma^2}{n})^{1-1/(2\delta)}$ | if $2\delta < 4(m+1)\beta+1$ |
| $i^{-2\beta}$ | $e^{-\kappa i}$ | $\frac{\sigma^2}{n}\lambda^{-1/2\beta}$ | $\lambda^{2(m+1)}$ | $(\frac{\sigma^2}{n})^{1/(2(m+1)+1/2\beta)}$ | $(\frac{\sigma^2}{n})^{1-1/(4(m+1)\beta+1)}$ | |
| $e^{-\rho i}$ | $i^{-2\delta}$ | $\frac{\sigma^2}{n}\log\frac{1}{\lambda}$ | $(\log\frac{1}{\lambda})^{1-2\delta}$ | $\exp(-(\frac{\sigma^2}{n})^{-1/(2\delta)})$ | $(\frac{\sigma^2}{n})^{1-1/(2\delta)}$ | |
| $e^{-\rho i}$ | $e^{-\kappa i}$ | $\frac{\sigma^2}{n}\log\frac{1}{\lambda}$ | $\lambda^{2(m+1)}$ | $(\frac{\sigma^2}{n})^{1/2}$ | $(\frac{\sigma^2}{n})\log(\frac{n}{\sigma^2})$ | if $\kappa > 2\rho$ |
| $e^{-\rho i}$ | $e^{-\kappa i}$ | $\frac{\sigma^2}{n}\log\frac{1}{\lambda}$ | $\lambda^{\kappa/\rho}$ | $(\frac{\sigma^2}{n})^{\rho/\kappa}$ | $(\frac{\sigma^2}{n})\log(\frac{n}{\sigma^2})$ | if $\kappa < 2\rho$ |

Table 1: Variance, bias, optimal regularization parameter, corresponding prediction performance, for several decays of eigenvalues and signal coefficients, and $m$-th order Richardson extrapolation (we always assume $\delta > 1/2$, $\beta > 1/2$, $\rho > 0$, $\kappa > 0$, to make the series summable). All entries are functions of $i$, $n$ or $\lambda$ and are only asymptotically bounded below and above, i.e., correspond to the asymptotic notation $\Theta(\cdot)$. Adapted from Bach (2013), changes due to potential Richardson extrapolation are highlighted in red.

In order to find the optimal regularization parameter, we minimize with respect to $\lambda$, that leads to $\lambda^{2(m+1)+1/2\beta} \propto (\sigma^2/n)$, leading to an optimal prediction performance proportional to $(\sigma^2/n)^\tau$, with $\tau = \frac{2(m+1)}{2(m+1)+\frac{1}{2\beta}} = 1 - \frac{\frac{1}{2\beta}}{2(m+1)+\frac{1}{2\beta}} = 1 - \frac{1}{4(m+1)\beta+1}$.

## E.2 Multiple extrapolation steps

Using $m$ steps of Richardson interpolation, we prove that we can get this regime as long as $2\delta < 4(m+1)\beta+1$, and thus with no limit if $m$ is large enough.

We thus consider

$$\hat{H}_\lambda^{(m)} = \sum_{i=1}^{m+1} \alpha_i^{(m)} \hat{H}_{i\lambda} = \sum_{i=1}^{m+1} \alpha_i^{(m)} K(K+n\lambda i I)^{-1} = I - n\lambda \sum_{i=1}^{m+1} i\alpha_i^{(m)} (K+n\lambda i I)^{-1},$$

using $\sum_{i=1}^{m+1} \alpha_i^{(m)} = 1$. We then use

$$\begin{aligned}
& (K+n\lambda i I)^{-1} \\
= \; & (K+n\lambda I)^{-1/2}\big[I + n\lambda(i-1)(K+\lambda I)^{-1}\big]^{-1}(K+n\lambda I)^{-1/2} \\
= \; & (K+n\lambda I)^{-1/2}\sum_{j=0}^{m-1}(-1)^j\big[n\lambda(K+n\lambda(i-1)I)^{-1}\big]^j(K+n\lambda I)^{-1/2} \\
& \qquad +(-1)^m(K+n\lambda I)^{-1/2}\big[n\lambda(i-1)(K+n\lambda I)^{-1}\big]^m\big[I+n\lambda(K+\lambda(i-1)I)^{-1}\big]^{-1}(K+n\lambda I)^{-1/2} \\
= \; & \sum_{j=0}^{m-1}(-1)^j(n\lambda)^j(K+n\lambda I)^{-j-1}(i-1)^j + (-1)^m(n\lambda)^m(K+n\lambda I)^{-m}(i-1)^m(K+n\lambda i I)^{-1},
\end{aligned}$$

where we have used the sum of a geometric series, with $A = n\lambda(i-1)(K+\lambda I)^{-1}$:

$$(I+A)^{-1} = \sum_{j=0}^{m-1}(-1)^j A^j + (-1)^m (I+A)^{-1}A^m.$$

Putting things together, we get

$$\begin{aligned}\hat{H}_\lambda^{(m)} - I &= -n\lambda \sum_{i=1}^{m+1} i\alpha_i^{(m)} \sum_{j=0}^{m-1} (-1)^j (n\lambda)^j (K+n\lambda I)^{-j-1}(i-1)^j \\ &\quad -n\lambda \sum_{i=1}^{m+1} i\alpha_i^{(m)} (-1)^m (n\lambda)^m (K+n\lambda I)^{-m}(i-1)^m (K+n\lambda i I)^{-1}.\end{aligned}$$

The first term is exactly zero by definition of the Richardson weights $\alpha_i^{(m)}$, and we are left with

$$(-1)^{m+1}\big[\hat{H}_\lambda^{(m)} - I\big] = (n\lambda)^{m+1} \sum_{i=1}^{m+1} i\alpha_i^{(m)} (i-1)^m (K+n\lambda I)^{-m}(K+n\lambda i I)^{-1}.$$

Thus

$$(n\lambda)^{-m-1}(K+n\lambda I)^{-m/2-1/2}\big[\hat{H}_\lambda^{(m)} - I\big](K+n\lambda I)^{-m/2-1/2}$$

has an operator norm bounded by a constant that depends on $m$ (and not on other quantities like $\lambda$ or $n$). This allows to bound the bias as

$$\mathrm{bias}(\hat{H}_\lambda^{(m)}) = \frac{1}{n}\|(\hat{H}_\lambda^{(m)} - I)z\|^2 \leqslant \square_m n^{2m+1}\lambda^{2m+2} z^\top (K+n\lambda I)^{-2m-2} z.$$

Similarly to the case $m=1$, $(\hat{H}_\lambda^{(m)})^2$ is upper-bounded by a sum of matrices which are all less than $K^2(K+n\lambda I)^{-2}$ (for the order between symmetric matrices), leading to a bound on the variance term as:

$$\mathrm{variance}(\hat{H}_\lambda^{(m)}) = \sigma^2 \operatorname{tr}[\hat{H}_\lambda^{(m)}]^2 \leqslant \triangle_m \frac{\sigma^2}{n} \operatorname{tr}[K(K+n\lambda I)^{-1}]^2.$$

Here $\square_m$ and $\triangle_m$ are constants that could be explicitly computed. As shown in the section, these constants have to diverge when $m$ tends to $+\infty$.

## E.3 Explicit expression

**Expression for $\alpha_i^{(m)}$.** We first give a proof for the explicit expression for $\alpha_i^{(m)}$. One approach is to solve Vandermonde matrices like done by Pagès (2007) in a similar context, but given the conjecture, we can simply check that it satisfies Eq. (1) and Eq. (2).

For Eq. (1), we have:

$$\sum_{i=1}^{m+1} (-1)^{i-1}\binom{m+1}{i} = 1 - \sum_{i=0}^{m+1} (-1)^i \binom{m+1}{i} = 0,$$

using the binomial formula. For Eq. (2), we have:

$$\sum_{i=1}^{m+1} (-1)^{i-1}\binom{m+1}{i} i = \sum_{i=1}^{m+1} (-1)^{i-1}(m+1)\binom{m}{i-1} = (m+1)\times 0 = 0,$$

and, more generally for any $j \in \{1,\dots,m\}$,

$$\sum_{i=j}^{m+1} (-1)^{i-1}\binom{m+1}{i} i(i-1)\cdots(i-j+1) = \sum_{i=j}^{m+1} (-1)^{i-1}(m+1)m\cdots(m-j+2)\binom{m-j+1}{i-j} = 0,$$

also using the binomial formula. This shows that $\sum_{i=j}^{m+1}(-1)^{i-1}\binom{m+1}{i} i^j = 0$ for all $j \in \{1,\dots,m\}$, which finishes the proof of the formula for $\alpha_i^{(m)}$.

**Expression for the smoothing matrix.** We can now provide an explicit expression for the extrapolated smoothing matrix. We have:

$$\begin{aligned}
\hat{H}_\lambda^{(m)} &= \sum_{i=1}^{m+1} \alpha_i^{(m)} \hat{H}_{i\lambda} \\
&= \sum_{i=1}^{m+1} \alpha_i^{(m)} K(K+n\lambda i I)^{-1} = I - n\lambda \sum_{i=1}^{m+1} i\alpha_i^{(m)} (K+n\lambda i I)^{-1} \\
&= I - n\lambda \sum_{i=1}^{m+1} i(-1)^{i-1} \binom{m+1}{i} (K+n\lambda i I)^{-1} = s(K/(n\lambda)),
\end{aligned}$$

where $s : \mathbb{R}^{n\times n} \to \mathbb{R}^{n\times n}$ is a spectral function defined on symmetric matrices from a function (note the classical overloaded notation) $s : \mathbb{R} \to \mathbb{R}$, by keeping eigenvectors unchanged and applying $s$ to eigenvalues (Golub and Van Loan, 1989, Chapter 11). We have, using a representation by an integral and the binomial formula:

$$\begin{aligned}
s(\mu) &= 1 - \sum_{i=1}^{m+1} i(-1)^{i-1} \binom{m+1}{i} \frac{1}{\mu+i} = 1 - \sum_{i=1}^{m+1} (m+1)(-1)^{i-1} \binom{m}{i-1} \int_0^1 t^{\mu+i-1} dt \\
&= 1 - (m+1) \int_0^1 t^\mu \Big[ \sum_{i=1}^{m+1} (-1)^{i-1} \binom{m}{i-1} t^{i-1} \Big] dt \\
&= 1 - (m+1) \int_0^1 t^\mu (1-t)^m dt = 1 - (m+1) \frac{\Gamma(1+\mu)\Gamma(1+m)}{\Gamma(2+m+\mu)},
\end{aligned}$$

using the expression of the Beta function in term of the Gamma function $\Gamma$ (Abramowitz et al., 1988). We can simply the expression as follows:

$$s(\mu) = 1 - \frac{(m+1)!}{(\mu+1)(\mu+2)\cdots(\mu+m+1)}.$$

This provides a new closed-form expression for Richardson extrapolation, as well as it provides an equivalent when $m$ tends to $+\infty$, as $s(\mu) \sim 1 - \frac{\Gamma(1+\mu)}{m^\mu}$. Therefore, when $m \to +\infty$, and $\mu > 0$, then $s(\mu)$ tends to one (which implies that the constants $\square_m$ and $\triangle_m$ cannot remain bounded). Therefore, the variance term converges to $\sigma^2$, but very slowly: the method does not blow up, but does not learn either.