Accelerated first-order methods

In the last lecture we provided convergence guarantees for gradient descent under two different assumptions. Under the stronger assumption that f was both M-smooth *and* strongly convex with parameter m, we showed that convergence to a tolerance of ϵ was possible in $O(\frac{M}{m}\log(1/\epsilon))$ iterations. Under the weaker assumption where we only assume that f is M-smooth, we were able to show that $O(M/\epsilon)$ iterations would be sufficient.

In this lecture we show that there are small changes we can make to gradient descent that can dramatically improve its performance, both in theory (resulting in improvements on the bounds above) and in practice. We will talk about two of these here: the heavy ball method and Nesterov's "optimal algorithm." Both of these strategies incorporate the idea of *momentum*, although in subtly different ways.

Momentum

One way to interpret gradient descent is as a discretization to the *gradient flow* differential equation

$$\begin{aligned} \boldsymbol{x}'(t) &= -\nabla f(\boldsymbol{x}(t)), \\ \boldsymbol{x}(0) &= \boldsymbol{x}_0. \end{aligned} \tag{1}$$

The solution to (1) is a curve that tracks the direction of steepest descent directly to the minimizer, where it arrives at a fixed point (where $\nabla f(\boldsymbol{x}) = \boldsymbol{0}$). To see how gradient descent arises as a discretization of (1), suppose we approximate the derivative with a forward difference

$$\boldsymbol{x}'(t) \approx \frac{\boldsymbol{x}(t+h) - \boldsymbol{x}(t)}{h},$$

for some small h. So if we think of $\boldsymbol{x}_{k+1}$ and $\boldsymbol{x}_k$ as closely spaced time points, we can interpret

$$\frac{1}{\alpha}\left(\boldsymbol{x}_{k+1} - \boldsymbol{x}_k\right) = -\nabla f(\boldsymbol{x}_k),$$

as a discrete approximation to gradient flow. Re-arranging the equation above yields the gradient descent iteration $\boldsymbol{x}_{k+1} = \boldsymbol{x}_k - \alpha \nabla f(\boldsymbol{x}_k)$.

The problem is once we perform this discretization, the path tends to oscillate. One way to get a more regular path is to consider an alternative differential equation that also has a fixed point where $\nabla f(\boldsymbol{x}) = 0$ but also incorporates a second-order term:

$$\mu \boldsymbol{x}''(t) + \boldsymbol{x}'(t) = -\nabla f(\boldsymbol{x}(t)). \tag{2}$$

From a physical perspective, this is a model for a particle with mass μ moving in a potential field with friction. This results in trajectories that develop momentum (a heavy ball will move down a hill faster than a light one in the presence of friction). In the case where $\mu = 0$ we recover (1), but in general the inclusion of the mass term above will result in a more accelerated trajectory towards the solution.

We can discretize the dynamics as before by setting

$$\boldsymbol{x}''(t) \approx \frac{\boldsymbol{x}_{k+1} - 2\boldsymbol{x}_k + \boldsymbol{x}_{k-1}}{h_1}, \quad \boldsymbol{x}'(t) \approx \frac{\boldsymbol{x}_k - \boldsymbol{x}_{k-1}}{h_2}.$$

If we plug these into (2) and rearrange we obtain an update rule of the form

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \beta_k(\boldsymbol{x}_k - \boldsymbol{x}_{k-1}) - \alpha_k \nabla f(\boldsymbol{x}_k), \tag{3}$$

where $\beta = h_1/h_2\mu$ and $\alpha = h_1/\mu$. This is the core iteration for the **heavy ball method**, introduced by Polyak in 1964 [Pol64]. The $\boldsymbol{x}_k - \boldsymbol{x}_{k-1}$ term above adds a little bit of the last step $\boldsymbol{x}_k - \boldsymbol{x}_{k-1}$ direction into the new step direction $\boldsymbol{x}_{k+1} - \boldsymbol{x}_k$ – this method is also referred to as *gradient descent with momentum*.

Convergence of the heavy ball method

In the previous lecture we showed that if $f(\boldsymbol{x})$ is M-smooth and strongly convex, then we can obtain a bound of the form

$$f(\boldsymbol{x}_{k+1}) - f(\boldsymbol{x}^\star) \leq \left(1 - \frac{1}{\kappa}\right)^k \left(f(\boldsymbol{x}_0) - f(\boldsymbol{x}^\star)\right),$$

where $\kappa = M/m$ is the "condition number." From this we can show that we can guarantee

$$\frac{f(\boldsymbol{x}_k) - f(\boldsymbol{x}^\star)}{f(\boldsymbol{x}_0) - f(\boldsymbol{x}^\star)} \leq \epsilon$$

provided that

$$k \geq \frac{\log(1/\epsilon)}{-\log(1 - 1/\kappa)}.$$

Using the inequality $\log(1 - x) \leq -x$ we can replace this with the simpler bound

$$k \geq \kappa \log\left(1/\epsilon\right).$$

In the technical details at the end of these notes we also provide an alternative argument for the convergence of gradient descent that begins by showing that

$$\|\boldsymbol{x}_k - \boldsymbol{x}^\star\|_2 \leq \left(\frac{\kappa - 1}{\kappa + 1}\right)^k \|\boldsymbol{x}_0 - \boldsymbol{x}^\star\|_2.$$

Using a similar argument as before, we can use this to show that

$$\frac{\|\boldsymbol{x}_k - \boldsymbol{x}^\star\|_2}{\|\boldsymbol{x}_0 - \boldsymbol{x}^\star\|_2} \leq \epsilon$$

provided that

$$k \gtrsim \kappa \log(1/\epsilon).$$

The heavy ball method significantly improves on this result in terms of its dependence on κ.

Specifically, under the same assumptions as before (M-smoothness and strong convexity), in the technical details we show that for the heavy ball method with

$$\alpha_k = \frac{4}{(\sqrt{M} - \sqrt{m})^2} \quad \text{and} \quad \beta_k = \frac{\sqrt{M} - \sqrt{m}}{\sqrt{M} + \sqrt{m}}$$

we have the bound

$$\|\boldsymbol{x}_k - \boldsymbol{x}^\star\|_2 \lesssim \left(\frac{\sqrt{\kappa} - 1}{\sqrt{\kappa} + 1}\right)^k \|\boldsymbol{x}_0 - \boldsymbol{x}^\star\|_2.$$

This can be translated into a guarantee that says

$$\frac{\|\boldsymbol{x}_k - \boldsymbol{x}^\star\|_2}{\|\boldsymbol{x}_0 - \boldsymbol{x}^\star\|_2} \leq \epsilon \quad \text{when} \quad k \gtrsim \sqrt{\kappa}\, \log(1/\epsilon).$$

The difference with gradient descent can be significant. When $\kappa = 10^2$, we are asking for $\approx 100 \log(1/\epsilon)$ iterations for gradient descent, as compared with $\approx 10 \log(1/\epsilon)$ from the heavy ball method.

Conjugate gradients

If you are familiar with the *method of conjugate gradients* (CG), some of this may feel vaguely familiar. If you have never heard of CG, I highly recommend reading through the tutorial "An introduction to the conjugate gradient method without the agonizing pain" [She94].

The CG method was developed for minimizing quadratic functions of the form $f(\boldsymbol{x}) = \frac{1}{2}\boldsymbol{x}^{\mathrm{T}}\boldsymbol{Q}\boldsymbol{x} - \boldsymbol{x}^{\mathrm{T}}\boldsymbol{b}$. While it is normally presented in quite a different fashion, it ultimately boils down to being a variant of the heavy ball method that is particularly well-suited to minimizing quadratic functions. To see this connection, note that the core CG iteration can be expressed[1] as

$$\begin{aligned} \boldsymbol{d}_k &= -\nabla f(\boldsymbol{x}_k) + \beta_k \boldsymbol{d}_{k-1} \\ \boldsymbol{x}_{k+1} &= \boldsymbol{x}_k + \alpha_k \boldsymbol{d}_k, \end{aligned}$$

where we start with $\boldsymbol{d}_0 = -\nabla f(\boldsymbol{x}_0)$. In CG, the β_k are set as

$$\beta_k = \frac{\|\nabla f(\boldsymbol{x}_k)\|_2^2}{\|\nabla f(\boldsymbol{x}_{k-1})\|_2^2}.$$

If $f(\boldsymbol{x})$ is a quadratic function this choice ensures that at each iteration $\boldsymbol{d}_k$ is *conjugate* to $\boldsymbol{d}_0, \ldots, \boldsymbol{d}_{k-1}$. We won't worry about saying more about this beyond the fact that this is a good idea *if* $f(\boldsymbol{x})$ *is quadratic*. Once β_k is fixed, α_k can then be chosen using a line search. Again, if $f(\boldsymbol{x})$ is quadratic, there is a simple closed form solution for this (which we have previously derived).

[1] You will typically see this algorithm described specifically for the quadratic case, in which case $\nabla f(\boldsymbol{x}) = Q\boldsymbol{x} - \boldsymbol{b}$ and these calculations are carefully broken up to re-use as many calculations as possible and avoid any unnecessary matrix-vector multiplies, so it may initially look quite different.

While CG is parameterized differently than the heavy ball method as described in (3), they are fundamentally the same. To see this note that we can also write

$$\begin{aligned}\boldsymbol{x}_{k+1} &= \boldsymbol{x}_k + \alpha_k \left(-\nabla f(\boldsymbol{x}_k) + \beta_k \boldsymbol{d}_{k-1}\right) \\ &= \boldsymbol{x}_k - \alpha_k \nabla f(\boldsymbol{x}_k) + \alpha_k \beta_k \frac{\boldsymbol{x}_k - \boldsymbol{x}_{k-1}}{\alpha_{k-1}}.\end{aligned}$$

This is precisely the same iteration as (3), but with a slightly different way of parameterizing the weight being applied to the momentum term.

If you are trying to minimize a quadratic function, CG is the way to go. The convergence guarantees you get for CG when minimizing a quadratic function are just as good (but not actually better than) what you have for the heavy ball method, but you don't need to know anything like Lipschitz or strong convexity parameters (which would correspond to the maximum and minimum eigenvalues of $\boldsymbol{Q}$) in order to choose the α_k and β_k.

However, if you are trying to minimize *anything else* CG is not necessarily a good choice. The choices for α_k and β_k are highly tuned to the quadratic setting and can yield unstable results in general.

Nesterov's "optimal" method

In the case where f is strictly convex, you can come up with examples that show that the convergence rate of the heavy ball method can't be improved in general. For non-strictly convex f, the story is more complicated.

Recall that we also have a convergence result for gradient descent in the case where we only assume M-smoothness. In particular, last time we showed that for a fixed step size $\alpha = 1/M$,

$$f(\boldsymbol{x}_k) - f(\boldsymbol{x}^\star) \leq \frac{M}{2k} \|\boldsymbol{x}_0 - \boldsymbol{x}^\star\|_2^2.$$

Thus, to reduce the error by a factor of ϵ requires

$$k \geq \frac{M}{2\epsilon}$$

iterations.

In 1983, Yuri Nesterov proposed a slight variation on the heavy ball method that can improve on this theory, and often works better in practice [Nes83].[2] Specifically, recall the heavy ball method, which can be represented via the iteration:

$$\begin{aligned} \boldsymbol{p}_k &= \beta_k (\boldsymbol{x}_k - \boldsymbol{x}_{k-1}) \\ \boldsymbol{x}_{k+1} &= \boldsymbol{x}_k + \boldsymbol{p}_k - \alpha_k \nabla f(\boldsymbol{x}_k), \end{aligned}$$

where we start with $\boldsymbol{p}_0 = \boldsymbol{0}$. Nesterov's method makes a subtle, but significant, change to this iteration:

$$\begin{aligned} \boldsymbol{p}_k &= \beta_k (\boldsymbol{x}_k - \boldsymbol{x}_{k-1}) \\ \boldsymbol{x}_{k+1} &= \boldsymbol{x}_k + \boldsymbol{p}_k - \alpha_k \nabla f(\boldsymbol{x}_k + \boldsymbol{p}_k). \end{aligned} \tag{4}$$

Notice that this is the same as heavy ball *except* that there is also a momentum term *inside* the gradient expression. With this iteration, we will show that (for a suitable choice of α_k and β_k)

$$f(\boldsymbol{x}_k) - f(\boldsymbol{x}^\star) \lesssim \frac{M}{k^2} \|\boldsymbol{x}_0 - \boldsymbol{x}^\star\|_2^2,$$

[2] Note that this method remained to a large extent unknown in the wider community until his 2004 publication (in English) of [Nes04].

meaning that we can reduce the error by a factor of ϵ in

$$k \gtrsim \frac{1}{\sqrt{\epsilon}},$$

iterations. When $\epsilon \sim 10^{-4}$, this is much, much better than $1/\epsilon$.

Nesterov's method is called "optimal" because it is impossible to beat the $1/k^2$ rate using only function and gradient evaluations. There are careful demonstrations of this in the literature (e.g., in [Nes04]).

Note that in practice, α_k can be chosen using a standard line search, and a good choice of β_k (both in practice, and as we will show below, in theory) turns out to be

$$\beta_k = \frac{k-1}{k+2}. \tag{5}$$

This tells us that we should initially not provide much weight to the momentum term, which makes intuitive sense as the initial gradients may not be pushing us in the right direction, but as we proceed we should have increased confidence that we are headed in the right direction and increase how much weight we place on the momentum term.

Significantly, note that in setting β_k we do *not* need to know anything about the function we are minimizing (such as strong convexity parameters). This represents an important advantage compared to the heavy ball method described above.

Convergence analysis of Nesterov's method

Analyzing the convergence of Nesterov's method under the assumption of M-smoothness is a little more involved than for gradient descent, but the overall approach is the same and contains many of the same elements, so we will start by recalling the main building blocks that we used in analyzing gradient descent.

Consequences of convexity and M-smoothness

First, we recall some basic facts that hold for any $\boldsymbol{x}, \boldsymbol{y} \in \operatorname{dom} f$. Since f is convex we have

$$f(\boldsymbol{y}) \geq f(\boldsymbol{x}) + \langle \boldsymbol{y} - \boldsymbol{x}, \nabla f(\boldsymbol{x}) \rangle. \tag{6}$$

Since f is M-smooth we have

$$f(\boldsymbol{y}) \leq f(\boldsymbol{x}) + \langle \boldsymbol{y} - \boldsymbol{x}, \nabla f(\boldsymbol{x}) \rangle + \frac{M}{2} \|\boldsymbol{y} - \boldsymbol{x}\|_2^2. \tag{7}$$

As a consequence of (7) (by setting $\boldsymbol{y} = \boldsymbol{x} - \frac{1}{M} \nabla f(\boldsymbol{x})$), we have that for any $\boldsymbol{x}$,

$$f\left(\boldsymbol{x} - \frac{\nabla f(\boldsymbol{x})}{M}\right) \leq f(\boldsymbol{x}) - \frac{\|\nabla f(\boldsymbol{x})\|_2^2}{2M}. \tag{8}$$

Combining this with the upper bound on $f(\boldsymbol{x})$ that you can obtain by rearranging (6), we obtain

$$f\left(\boldsymbol{x} - \frac{\nabla f(\boldsymbol{x})}{M}\right) \leq f(\boldsymbol{y}) + \langle \boldsymbol{x} - \boldsymbol{y}, \nabla f(\boldsymbol{x}) \rangle - \frac{\|\nabla f(\boldsymbol{x})\|_2^2}{2M}. \tag{9}$$

As we will see below, this inequality is the foundation of our analysis of both gradient descent and Nesterov's method. By plugging in different choices for $\boldsymbol{y}$ (such as $\boldsymbol{x}_k$ or $\boldsymbol{x}^\star$) we can obtain both *lower*

bounds on how much progress we make when we take a gradient step as well as *upper* bounds on how far away we are from a global optimum.

Convergence of gradient descent

Recall that in our analysis for gradient we assume a fixed step size $\alpha = 1/M$, resulting in an update rule of

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k - \frac{\nabla f(\boldsymbol{x}_k)}{M}.$$

Thus, setting $\boldsymbol{x} = \boldsymbol{x}_k$ and $\boldsymbol{y} = \boldsymbol{x}^\star$ in (9) implies that

$$f(\boldsymbol{x}_{k+1}) \leq f(\boldsymbol{x}^\star) + M\langle \boldsymbol{x}_k - \boldsymbol{x}^\star, \boldsymbol{x}_k - \boldsymbol{x}_{k+1}\rangle - \frac{M}{2}\|\boldsymbol{x}_k - \boldsymbol{x}_{k+1}\|_2^2.$$

From this, if we define $\delta_k = f(\boldsymbol{x}_k) - f(\boldsymbol{x}^\star)$ and do some algebraic manipulation (see the previous notes) we get a bound of the form

$$\delta_{k+1} \leq \frac{M}{2}\left(\|\boldsymbol{x}_k - \boldsymbol{x}^\star\|_2^2 - \|\boldsymbol{x}_{k+1} - \boldsymbol{x}^\star\|_2^2\right).$$

This yields the *telescopic sum*

$$\begin{aligned}
\sum_{i=0}^{k-1} \delta_{i+1} &\leq \frac{M}{2}\left(\sum_{i=0}^{k-1} \|\boldsymbol{x}_i - \boldsymbol{x}^\star\|_2^2 - \|\boldsymbol{x}_{i+1} - \boldsymbol{x}^\star\|_2^2\right) \\
&= \frac{M}{2}\left(\|\boldsymbol{x}_0 - \boldsymbol{x}^\star\|_2^2 - \|\boldsymbol{x}_k - \boldsymbol{x}^\star\|_2^2\right) \\
&\leq \frac{M}{2}\|\boldsymbol{x}_0 - \boldsymbol{x}^\star\|_2^2.
\end{aligned}$$

The proof for gradient descent concludes by noting that

$$\delta_k \leq \frac{1}{k}\sum_{i=0}^{k-1} \delta_{i+1} \leq \frac{M}{2k}\|\boldsymbol{x}_0 - \boldsymbol{x}^\star\|_2^2.$$

Convergence of Nesterov's method

We will follow a similar argument to analyze Nesterov's method. We will again take $\alpha_k = 1/M$, but we will see that the analysis suggests a natural choice for β_k. With this choice of α_k, the main iteration from (4) is

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \boldsymbol{p}_k - \frac{1}{M}\nabla f(\boldsymbol{x}_k + \boldsymbol{p}_k).$$

It will be convenient to define

$$\boldsymbol{g}_k = -\frac{1}{M}\nabla f(\boldsymbol{x}_k + \boldsymbol{p}_k),$$

so that the main iteration becomes simply $\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \boldsymbol{p}_k + \boldsymbol{g}_k$. With this notation, by setting $\boldsymbol{x} = \boldsymbol{x}_k + \boldsymbol{p}_k$ in (9) we obtain the bound

$$f(\boldsymbol{x}_{k+1}) \leq f(\boldsymbol{y}) - M\langle \boldsymbol{x}_k - \boldsymbol{p}_k - \boldsymbol{y}, \boldsymbol{g}_k\rangle - \frac{M}{2}\|\boldsymbol{g}_k\|_2^2. \tag{10}$$

If we set $\boldsymbol{y} = \boldsymbol{x}^\star$ in (10) and again let δ_k denote $f(\boldsymbol{x}_k) - f(\boldsymbol{x}^\star)$ we obtain

$$\delta_{k+1} \leq \frac{M}{2}\left(2\langle \boldsymbol{x}^\star - \boldsymbol{x}_k - \boldsymbol{p}_k, \boldsymbol{g}_k\rangle - \|\boldsymbol{g}_k\|_2^2\right). \tag{11}$$

In our analysis of gradient descent, we then tried to rearrange an analogous bound to obtain a telescopic sum, but that doesn't quite work here. Instead we will need to combine (11) with another bound. Noting that $\delta_k - \delta_{k+1} = f(\boldsymbol{x}_k) - f(\boldsymbol{x}_{k+1})$, we observe that setting $\boldsymbol{y} = \boldsymbol{x}_k$ in (10) yields

$$\delta_k - \delta_{k+1} \geq \frac{M}{2}\left(2\langle \boldsymbol{p}_k, \boldsymbol{g}_k\rangle + \|\boldsymbol{g}_k\|_2^2\right). \tag{12}$$

We now consider the inequality formed by adding together (11) and $1 - \lambda_k$ times (12) (where λ_k is something we will choose later, but

satisfies $\lambda_k \geq 1$, so that this multiplication switches the direction of the inequality). The left-hand side of the sum will be

$$\delta_{k+1} + (1-\lambda_k)(\delta_k - \delta_{k+1}) = \lambda_k \delta_{k+1} - (\lambda_k - 1)\delta_k.$$

The right-hand side of the sum will be

$$\begin{aligned}
&\frac{M}{2}\left(2\langle \boldsymbol{x}^\star - \boldsymbol{x}_k - \boldsymbol{p}_k + (1-\lambda_k)\boldsymbol{p}_k, \boldsymbol{g}_k\rangle - \|\boldsymbol{g}_k\|_2^2 + (1-\lambda_k)\|\boldsymbol{g}_k\|_2^2\right) \\
&\quad = \frac{M}{2}\left(2\langle \boldsymbol{x}^\star - \boldsymbol{x}_k - \lambda_k\boldsymbol{p}_k, \boldsymbol{g}_k\rangle - \lambda_k\|\boldsymbol{g}_k\|_2^2\right) \\
&\quad = \frac{M}{2\lambda_k}\left(2\langle \boldsymbol{x}^\star - \boldsymbol{x}_k - \lambda_k\boldsymbol{p}_k, \lambda_k\boldsymbol{g}_k\rangle - \|\lambda_k\boldsymbol{g}_k\|_2^2\right) \\
&\quad = \frac{M}{2\lambda_k}\left(\|\boldsymbol{x}^\star - \boldsymbol{x}_k - \lambda_k\boldsymbol{p}_k\|_2^2 - \|\boldsymbol{x}^\star - \boldsymbol{x}_k - \lambda_k\boldsymbol{p}_k - \lambda_k\boldsymbol{g}_k\|_2^2\right),
\end{aligned}$$

where the last equality follows from the easily verified fact that $2\langle \boldsymbol{a}, \boldsymbol{b}\rangle - \|\boldsymbol{b}\|_2^2 = \|\boldsymbol{a}\|_2^2 - \|\boldsymbol{a} - \boldsymbol{b}\|_2^2$. If we make the substitution $\boldsymbol{u}_k = \boldsymbol{x}_k + \lambda_k\boldsymbol{p}_k$, then combining these yields the inequality

$$\lambda_k^2\delta_{k+1} - (\lambda_k^2 - \lambda_k)\delta_k \leq \frac{M}{2}\left(\|\boldsymbol{x}^\star - \boldsymbol{u}_k\|_2^2 - \|\boldsymbol{x}^\star - \boldsymbol{u}_k - \lambda_k\boldsymbol{g}_k\|_2^2\right). \tag{13}$$

We will now show that if we choose λ_k and β_k appropriately, (13) will yield a telescopic sum on both sides. This will occur on right-hand side of (13) if

$$\boldsymbol{u}_k + \lambda_k\boldsymbol{g}_k = \boldsymbol{u}_{k+1}.$$

Noting that $\boldsymbol{p}_{k+1} = \beta_{k+1}(\boldsymbol{x}_{k+1} - \boldsymbol{x}_k) = \beta_{k+1}(\boldsymbol{p}_k + \boldsymbol{g}_k)$, we can write

$$\begin{aligned}
\boldsymbol{u}_{k+1} &= \boldsymbol{x}_{k+1} + \lambda_{k+1}\boldsymbol{p}_{k+1} \\
&= \boldsymbol{x}_k + \boldsymbol{p}_k + \boldsymbol{g}_k + \lambda_{k+1}\beta_{k+1}(\boldsymbol{p}_k + \boldsymbol{g}_k) \\
&= \boldsymbol{x}_k + (1 + \lambda_{k+1}\beta_{k+1})(\boldsymbol{p}_k + \boldsymbol{g}_k).
\end{aligned}$$

Thus, to make $\boldsymbol{u}_{k+1}$ equal to $\boldsymbol{u}_k + \lambda_k \boldsymbol{g}_k = \boldsymbol{x}_k + \lambda_k(\boldsymbol{p}_k + \boldsymbol{g}_k)$ we simply need to have

$$\lambda_k = 1 + \lambda_{k+1}\beta_{k+1} \Rightarrow \beta_{k+1} = \frac{\lambda_k - 1}{\lambda_{k+1}}. \tag{14}$$

For β_k satisfying (14), if we sum (13) from $i = 0$ to $k-1$ we thus have

$$\begin{aligned}
\sum_{i=0}^{k-1} \lambda_i^2 \delta_{i+1} - (\lambda_i^2 - \lambda_i)\delta_i &\leq \frac{M}{2}\left(\|\boldsymbol{x}^\star - \boldsymbol{u}_0\|_2^2 - \|\boldsymbol{x}^\star - \boldsymbol{u}_k\|_2^2\right) \\
&\leq \frac{M}{2}\|\boldsymbol{x}^\star - \boldsymbol{u}_0\|_2^2 \\
&= \frac{M}{2}\|\boldsymbol{x}^\star - \boldsymbol{x}_0\|_2^2.
\end{aligned} \tag{15}$$

Next, one possible approach is to choose the λ_k so as to obtain a telescopic sum on the left-hand side of the inequality as well. This is the approach you will see most often in analyzing the convergence of Nesterov's method, but it is a little involved and leads to a recursive formula for λ_k (and hence β_k) instead of a simple closed form expression. Instead we will choose a simpler λ_k that yields essentially the same bound.

Specifically, suppose that we set $\lambda_k = (k+2)/2$. First, note that from (14) this yields

$$\beta_{k+1} = \frac{\frac{k+2}{2} - 1}{\frac{k+1}{2}} = \frac{k}{k+3},$$

which coincides with the rule for setting β_k given in (5). Next, note that we can write

$$\sum_{i=0}^{k-1} \lambda_i^2 \delta_{i+1} - (\lambda_i^2 - \lambda_i)\delta_i = (\lambda_0 - \lambda_0^2)\delta_0 + \lambda_{k-1}^2 \delta_k + \sum_{i=1}^{k-1} (\lambda_{i-1}^2 - \lambda_i^2 + \lambda_i)\delta_i.$$

Plugging in $\lambda_i = (i+2)/2$ yields

$$\sum_{i=0}^{k-1} \lambda_i^2 \delta_{i+1} - (\lambda_i^2 - \lambda_i)\delta_i = \left(\frac{k+1}{2}\right)^2 \delta_k + \frac{1}{4}\sum_{i=0}^{k-1} \delta_i$$
$$\geq \left(\frac{k+1}{2}\right)^2 \delta_k,$$

where the inequality follows since $\delta_i = f(\boldsymbol{x}_i) - f(\boldsymbol{x}^\star) \geq 0$. Combining this lower bound with (15) yields

$$\left(\frac{k+1}{2}\right)^2 \delta_k \leq \frac{M}{2}\|\boldsymbol{x}^\star - \boldsymbol{x}_0\|_2^2$$

or equivalently

$$f(\boldsymbol{x}_k) - f(\boldsymbol{x}^\star) \leq \frac{2M}{(k+1)^2}\|\boldsymbol{x}^\star - \boldsymbol{x}_0\|_2^2,$$

which is exactly the $O(1/k^2)$ convergence rate we wanted.

Technical Details: Analysis of the heavy ball method

In analyzing the heavy ball method we will assume that $f(\boldsymbol{x})$ is both M-smooth and strongly convex.

Moreover, here we will also assume that $f(\boldsymbol{x})$ is twice differentiable, in which case these assumptions can be captured simply as:

$$m\mathbf{I} \preceq \nabla^2 f(\boldsymbol{x}) \preceq M\mathbf{I}, \quad \text{for all} \;\; \boldsymbol{x} \in \operatorname{dom} f. \tag{16}$$

This means that the eigenvalues of the Hessian matrix are uniformly bounded between m and M at all points $\boldsymbol{x}$.

Our approach to analyzing the heavy ball method will be somewhat different to the one we took in our initial analysis of gradient descent. Recall that before we used the PL inequality to obtain the bound

$$f(\boldsymbol{x}_{k+1}) - f(\boldsymbol{x}^\star) \leq \frac{M-m}{M} \left(f(\boldsymbol{x}_k) - f(\boldsymbol{x}^\star) \right).$$

When $f(\boldsymbol{x})$ is twice differentiable there is an alternative approach that results in the similar bound

$$\|\boldsymbol{x}_{k+1} - \boldsymbol{x}^\star\|_2 \leq \frac{M-m}{M+m} \|\boldsymbol{x}_k - \boldsymbol{x}^\star\|_2.$$

Our approach to the heavy ball method will be an extension of this, so we will first see how this works in the simpler case of gradient descent.

The analysis for gradient descent uses (16) with an application of the Taylor theorem. Briefly, we have

$$\begin{aligned}\|\boldsymbol{x}_{k+1} - \boldsymbol{x}^\star\|_2 &= \|\boldsymbol{x}_k - \alpha_k \nabla f(\boldsymbol{x}_k) - \boldsymbol{x}^\star\|_2 \\ &= \|\boldsymbol{x}_k - \boldsymbol{x}^\star - \alpha_k \left(\nabla f(\boldsymbol{x}_k) - \nabla f(\boldsymbol{x}^\star)\right)\|_2 \\ &= \left\|\left(\mathbf{I} - \alpha_k \nabla^2 f(\boldsymbol{z})\right)(\boldsymbol{x}_k - \boldsymbol{x}^\star)\right\|_2 \\ &\leq \left\|\mathbf{I} - \alpha_k \nabla^2 f(\boldsymbol{z})\right\| \cdot \|\boldsymbol{x}_k - \boldsymbol{x}^\star\|_2,\end{aligned}$$

where the second equality comes from the fact that $\nabla f(\boldsymbol{x}^\star) = \mathbf{0}$, while the third equality comes from the Taylor theorem: there exists some $\boldsymbol{z}$ on the line between $\boldsymbol{x}_k$ and $\boldsymbol{x}^\star$ such that

$$\nabla^2 f(\boldsymbol{z})(\boldsymbol{x}_k - \boldsymbol{x}^\star) = \nabla f(\boldsymbol{x}_k) - \nabla f(\boldsymbol{x}^\star).$$

Since we have a bound on the eigenvalues of $\nabla^2 f(\boldsymbol{z})$, we know that the maximum eigenvalue of the symmetric matrix $\mathbf{I} - \alpha_k \nabla^2 f(\boldsymbol{z})$ is no more than

$$\left\|\mathbf{I} - \alpha_k \nabla^2 f(\boldsymbol{z})\right\| \;\leq\; \max\left(|1 - \alpha_k m|, |1 - \alpha_k M|\right).$$

If we take $\alpha_k = 2/(M+m)$, we obtain

$$\left\|\mathbf{I} - \alpha_k \nabla^2 f(\boldsymbol{z})\right\| \;\leq\; \frac{M-m}{M+m} = \frac{\kappa - 1}{\kappa + 1},$$

where $\kappa = M/m$ is the "condition number". So we have

$$\|\boldsymbol{x}_{k+1} - \boldsymbol{x}^\star\|_2 \leq \left(\frac{\kappa - 1}{\kappa + 1}\right) \|\boldsymbol{x}_k - \boldsymbol{x}^\star\|_2,$$

which, by induction on k means

$$\|\boldsymbol{x}_k - \boldsymbol{x}^\star\|_2 \leq \left(\frac{\kappa - 1}{\kappa + 1}\right)^k \|\boldsymbol{x}_0 - \boldsymbol{x}^\star\|_2.$$

For the heavy ball method, we have a similar analysis that ends in a better result. We start by looking at how $\|\boldsymbol{x}_{k+1} - \boldsymbol{x}^\star\|^2 + \|\boldsymbol{x}_k - \boldsymbol{x}^\star\|_2^2$ goes to zero for fixed values of α, β which we will choose later:

$$\begin{aligned}
\left\| \begin{bmatrix} \boldsymbol{x}_{k+1} - \boldsymbol{x}^\star \\ \boldsymbol{x}_k - \boldsymbol{x}^\star \end{bmatrix} \right\|_2 &= \left\| \begin{bmatrix} \boldsymbol{x}_k + \beta(\boldsymbol{x}_k - \boldsymbol{x}_{k-1}) - \boldsymbol{x}^\star \\ \boldsymbol{x}_k - \boldsymbol{x}^\star \end{bmatrix} - \alpha \begin{bmatrix} \nabla f(\boldsymbol{x}_k) - \nabla f(\boldsymbol{x}^\star) \\ \mathbf{0} \end{bmatrix} \right\|_2 \\
&= \left\| \begin{bmatrix} \boldsymbol{x}_k + \beta(\boldsymbol{x}_k - \boldsymbol{x}_{k-1}) - \boldsymbol{x}^\star \\ \boldsymbol{x}_k - \boldsymbol{x}^\star \end{bmatrix} - \alpha \begin{bmatrix} \nabla^2 f(\boldsymbol{z})\,(\boldsymbol{x}_k - \boldsymbol{x}^\star) \\ \mathbf{0} \end{bmatrix} \right\|_2 \\
&= \left\| \begin{bmatrix} (1+\beta)\mathbf{I} - \alpha\nabla^2 f(\boldsymbol{z}) & -\beta\mathbf{I} \\ \mathbf{I} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{x}_k - \boldsymbol{x}^\star \\ \boldsymbol{x}_{k-1} - \boldsymbol{x}^\star \end{bmatrix} \right\|_2.
\end{aligned}$$

Applying the above iteratively means that, in the limit

$$\|\boldsymbol{x}_k - \boldsymbol{x}^\star\|_2 \;\lesssim\; \|\boldsymbol{T}^k\|\,\|\boldsymbol{x}_0 - \boldsymbol{x}^\star\|_2.$$

where

$$\boldsymbol{T} = \begin{bmatrix} (1+\beta)\mathbf{I} - \alpha\nabla^2 f(\boldsymbol{z}) & -\beta\mathbf{I} \\ \mathbf{I} & \mathbf{0} \end{bmatrix}.$$

It is a fundamental result from numerical linear algebra that as k gets large, $\|\boldsymbol{T}^k\|$ becomes very close to[3] $\rho(\boldsymbol{T})^k$, where ρ is the maximum of the magnitudes of the eigenvalues of $\boldsymbol{T}$.

We can get at these eigenvalues in a systematic way. Start by taking an eigenvalue decomposition of the Hessian matrix above, $\nabla^2 f(\boldsymbol{z}) = \boldsymbol{V}\boldsymbol{\Lambda}\boldsymbol{V}^{\mathrm{T}}$. Since $\boldsymbol{V}\boldsymbol{V}^{\mathrm{T}} = \mathbf{I}$, we can write

$$\begin{bmatrix} (1+\beta)\mathbf{I} - \alpha\nabla^2 f(\boldsymbol{z}) & -\beta\mathbf{I} \\ \mathbf{I} & \mathbf{0} \end{bmatrix} = \begin{bmatrix} \boldsymbol{V} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{V} \end{bmatrix} \begin{bmatrix} (1+\beta)\mathbf{I} - \alpha\boldsymbol{\Lambda} & -\beta\mathbf{I} \\ \mathbf{I} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{V}^{\mathrm{T}} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{V}^{\mathrm{T}} \end{bmatrix}.$$

Since $\begin{bmatrix} \boldsymbol{V} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{V} \end{bmatrix}$ is orthonormal, its application on the left and the right of a matrix does not change the magnitude of its eigenvalues,

[3] If $\boldsymbol{T}$ is symmetric, then of course $\|\boldsymbol{T}^k\| = \rho(\boldsymbol{T})^k$ for all k.

and we have

$$\begin{aligned}\rho(\boldsymbol{T}) &= \rho\left(\begin{bmatrix}(1+\beta)\mathbf{I} - \alpha\boldsymbol{\Lambda} & -\beta\mathbf{I} \\ \mathbf{I} & \mathbf{0}\end{bmatrix}\right) \\ &= \max_{n=1,\ldots,N} \rho(\boldsymbol{T}_n),\end{aligned}$$

where

$$\boldsymbol{T}_n = \begin{bmatrix}1+\beta-\alpha\lambda_n & -\beta \\ 1 & 0\end{bmatrix}.$$

This last equality follows from the fact the large $2N \times 2N$ matrix can have its rows and columns permuted (which doesn't change the eigenvalues) to become block diagonal, with the 2×2 matrices $\boldsymbol{T}_n$ as the blocks.

We now have the problem of finding α, β that minimize the size of the largest eigenvalue of the 2×2 matrices above given the knowledge that $m \leq \lambda_n \leq M$. A very technical calculation yields the fact that taking

$$\alpha = \frac{4}{(\sqrt{M}-\sqrt{m})^2}, \quad \beta = \frac{\sqrt{M}-\sqrt{m}}{\sqrt{M}+\sqrt{m}},$$

yields

$$\rho(\boldsymbol{T}_n) \leq \frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}, \quad \text{for all } n.$$

Thus the convergence of the heavy ball method is approximately

$$\|\boldsymbol{x}_k - \boldsymbol{x}^\star\|_2 \lesssim \left(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right)^k \|\boldsymbol{x}_0 - \boldsymbol{x}^\star\|_2.$$

References

[Nes83] Y. Nesterov. A method of solving a convex programming problem with convergence rate $O(1/k^2)$. *Proc. USSR Acad. Sci.*, 269:543–547, 1983.

[Nes04] Y. Nesterov. *Introductory Lectures on Convex Optimization*. Springer Science+Business Media, 2004.

[Pol64] B. Polyak. Some methods of speeding up the convergence of iteration methods. *USSR Computational Mathematics and Mathematical Physics*, 4(5):1–17, 1964.

[She94] J. Shewchuk. An introduction to the conjugate gradient method without the agonizing pain. 1994.