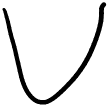
Discover acceleration of gradient descent

Daniil Merkulov

Optimization for ML. Faculty of Computer Science. HSE University

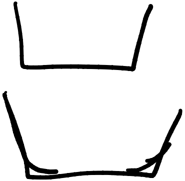
Previously

Gradient Descent: $\min\limits_{x \in \mathbb{R}^n} f(x)$ $\qquad x^{k+1} = x^k - \alpha^k \nabla f(x^k)$



ε

convex (non-smooth)	smooth (non-convex)	smooth & convex	smooth & strongly convex (or PL)
$f(x^k) - f^* \sim \mathcal{O}\left(\dfrac{1}{\sqrt{k}}\right)$	$\\|\nabla f(x^k)\\|^2 \sim \mathcal{O}\left(\dfrac{1}{k}\right)$	$f(x^k) - f^* \sim \mathcal{O}\left(\dfrac{1}{k}\right)$	$\\|x^k - x^*\\|^2 \sim \mathcal{O}\left(\left(1 - \dfrac{\mu}{L}\right)^k\right)$
$k_\varepsilon \sim \mathcal{O}\left(\dfrac{1}{\varepsilon^2}\right)$	$k_\varepsilon \sim \mathcal{O}\left(\dfrac{1}{\varepsilon}\right)$	$k_\varepsilon \sim \mathcal{O}\left(\dfrac{1}{\varepsilon}\right)$	$k_\varepsilon \sim \mathcal{O}\left(\kappa \log \dfrac{1}{\varepsilon}\right)$



$k \sim \dfrac{1}{\varepsilon}$

Previously

Gradient Descent: $\min\limits_{x \in \mathbb{R}^n} f(x)$ $\qquad x^{k+1} = x^k - \alpha^k \nabla f(x^k)$

convex (non-smooth)	smooth (non-convex)	smooth & convex	smooth & strongly convex (or PL)
$f(x^k) - f^* \sim \mathcal{O}\left(\dfrac{1}{\sqrt{k}}\right)$	$\|\nabla f(x^k)\|^2 \sim \mathcal{O}\left(\dfrac{1}{k}\right)$	$f(x^k) - f^* \sim \mathcal{O}\left(\dfrac{1}{k}\right)$	$\|x^k - x^*\|^2 \sim \mathcal{O}\left(\left(1 - \dfrac{\mu}{L}\right)^k\right)$
$k_\varepsilon \sim \mathcal{O}\left(\dfrac{1}{\varepsilon^2}\right)$	$k_\varepsilon \sim \mathcal{O}\left(\dfrac{1}{\varepsilon}\right)$	$k_\varepsilon \sim \mathcal{O}\left(\dfrac{1}{\varepsilon}\right)$	$k_\varepsilon \sim \mathcal{O}\left(\kappa \log \dfrac{1}{\varepsilon}\right)$

For smooth strongly convex we have:

$$
f(x^k) - f^* \leq \left(1 - \frac{\mu}{L}\right)^k (f(x^0) - f^*).
$$

Note also, that for any x

$$
1 - x \leq e^{-x}
$$

Previously

Gradient Descent: $\min\limits_{x \in \mathbb{R}^n} f(x)$ $\qquad x^{k+1} = x^k - \alpha^k \nabla f(x^k)$

convex (non-smooth)	smooth (non-convex)	smooth & convex	smooth & strongly convex (or PL)
$f(x^k) - f^* \sim \mathcal{O}\left(\frac{1}{\sqrt{k}}\right)$	$\|\nabla f(x^k)\|^2 \sim \mathcal{O}\left(\frac{1}{k}\right)$	$f(x^k) - f^* \sim \mathcal{O}\left(\frac{1}{k}\right)$	$\|x^k - x^*\|^2 \sim \mathcal{O}\left(\left(1 - \frac{\mu}{L}\right)^k\right)$
$k_\varepsilon \sim \mathcal{O}\left(\frac{1}{\varepsilon^2}\right)$	$k_\varepsilon \sim \mathcal{O}\left(\frac{1}{\varepsilon}\right)$	$k_\varepsilon \sim \mathcal{O}\left(\frac{1}{\varepsilon}\right)$	$k_\varepsilon \sim \mathcal{O}\left(\kappa \log \frac{1}{\varepsilon}\right)$

For smooth strongly convex we have:

$$f(x^k) - f^* \leq \left(1 - \frac{\mu}{L}\right)^k (f(x^0) - f^*).$$

Note also, that for any x

$$1 - x \leq e^{-x}$$

Finally we have

$$\varepsilon = f(x^{k_\varepsilon}) - f^* \leq \left(1 - \frac{\mu}{L}\right)^{k_\varepsilon} (f(x^0) - f^*)$$

$$\leq \exp\left(-k_\varepsilon \frac{\mu}{L}\right) (f(x^0) - f^*)$$

$$k_\varepsilon \geq \kappa \log \frac{f(x^0) - f^*}{\varepsilon} = \mathcal{O}\left(\kappa \log \frac{1}{\varepsilon}\right)$$

$\varkappa = \frac{L}{\mu} > 1$

Previously

Gradient Descent: $$\min_{x \in \mathbb{R}^n} f(x) \qquad x^{k+1} = x^k - \alpha^k \nabla f(x^k)$$

convex (non-smooth)	smooth (non-convex)	smooth & convex	smooth & strongly convex (or PL)
$f(x^k) - f^* \sim \mathcal{O}\left(\frac{1}{\sqrt{k}}\right)$ $k_\varepsilon \sim \mathcal{O}\left(\frac{1}{\varepsilon^2}\right)$	$\|\nabla f(x^k)\|^2 \sim \mathcal{O}\left(\frac{1}{k}\right)$ $k_\varepsilon \sim \mathcal{O}\left(\frac{1}{\varepsilon}\right)$	$f(x^k) - f^* \sim \mathcal{O}\left(\frac{1}{k}\right)$ $k_\varepsilon \sim \mathcal{O}\left(\frac{1}{\varepsilon}\right)$	$\|x^k - x^*\|^2 \sim \mathcal{O}\left(\left(1 - \frac{\mu}{L}\right)^k\right)$ $k_\varepsilon \sim \mathcal{O}\left(\kappa \log \frac{1}{\varepsilon}\right)$

For smooth strongly convex we have:

$$f(x^k) - f^* \leq \left(1 - \frac{\mu}{L}\right)^k (f(x^0) - f^*).$$

Note also, that for any x

$$1 - x \leq e^{-x}$$

Finally we have

$$\begin{aligned}
\varepsilon = f(x^{k_\varepsilon}) - f^* &\leq \left(1 - \frac{\mu}{L}\right)^{k_\varepsilon} (f(x^0) - f^*) \\
&\leq \exp\left(-k_\varepsilon \frac{\mu}{L}\right) (f(x^0) - f^*) \\
k_\varepsilon \geq \kappa \log \frac{f(x^0) - f^*}{\varepsilon} &= \mathcal{O}\left(\kappa \log \frac{1}{\varepsilon}\right)
\end{aligned}$$

Question: Can we do faster, than this using the first-order information?

Previously

Gradient Descent: $\min\limits_{x \in \mathbb{R}^n} f(x) \qquad x^{k+1} = x^k - \alpha^k \nabla f(x^k)$

convex (non-smooth)	smooth (non-convex)	smooth & convex	smooth & strongly convex (or PL)
$f(x^k) - f^* \sim \mathcal{O}\left(\dfrac{1}{\sqrt{k}}\right)$	$\\|\nabla f(x^k)\\|^2 \sim \mathcal{O}\left(\dfrac{1}{k}\right)$	$f(x^k) - f^* \sim \mathcal{O}\left(\dfrac{1}{k}\right)$	$\\|x^k - x^*\\|^2 \sim \mathcal{O}\left(\left(1 - \dfrac{\mu}{L}\right)^k\right)$
$k_\varepsilon \sim \mathcal{O}\left(\dfrac{1}{\varepsilon^2}\right)$	$k_\varepsilon \sim \mathcal{O}\left(\dfrac{1}{\varepsilon}\right)$	$k_\varepsilon \sim \mathcal{O}\left(\dfrac{1}{\varepsilon}\right)$	$k_\varepsilon \sim \mathcal{O}\left(\kappa \log \dfrac{1}{\varepsilon}\right)$

For smooth strongly convex we have:

$$f(x^k) - f^* \leq \left(1 - \frac{\mu}{L}\right)^k (f(x^0) - f^*).$$

Note also, that for any x

$$1 - x \leq e^{-x}$$

Finally we have

$$\begin{aligned}
\varepsilon = f(x^{k_\varepsilon}) - f^* &\leq \left(1 - \frac{\mu}{L}\right)^{k_\varepsilon} (f(x^0) - f^*) \\
&\leq \exp\left(-k_\varepsilon \frac{\mu}{L}\right) (f(x^0) - f^*) \\
k_\varepsilon \geq \kappa \log \frac{f(x^0) - f^*}{\varepsilon} &= \mathcal{O}\left(\kappa \log \frac{1}{\varepsilon}\right)
\end{aligned}$$

Question: Can we do faster, than this using the first-order information? **Yes, we can.**

Lower bounds

GD $\frac{1}{k}$ $\frac{1}{k^2}$?

convex (non-smooth)	smooth (non-convex)[1]	smooth & convex[2]	smooth & strongly convex (or PL)
$\mathcal{O}\left(\frac{1}{\sqrt{k}}\right)$	$\mathcal{O}\left(\frac{1}{k^2}\right)$	$\mathcal{O}\left(\frac{1}{k^2}\right)$	$\mathcal{O}\left(\left(1-\sqrt{\frac{\mu}{L}}\right)^k\right)$
$k_\varepsilon \sim \mathcal{O}\left(\frac{1}{\varepsilon^2}\right)$	$k_\varepsilon \sim \mathcal{O}\left(\frac{1}{\sqrt{\varepsilon}}\right)$	$k_\varepsilon \sim \mathcal{O}\left(\frac{1}{\sqrt{\varepsilon}}\right)$	$k_\varepsilon \sim \mathcal{O}\left(\sqrt{\kappa}\log\frac{1}{\varepsilon}\right)$

[1] Carmon, Duchi, Hinder, Sidford, 2017
[2] Nemirovski, Yudin, 1979

Lower bounds

The iteration of gradient descent:

$$
\begin{aligned}
x^{k+1} &= x^k - \alpha^k \nabla f(x^k) \\
&= x^{k-1} - \alpha^{k-1} \nabla f(x^{k-1}) - \alpha^k \nabla f(x^k) \\
&\;\vdots \\
&= x^0 - \sum_{i=0}^{k} \alpha^{k-i} \nabla f(x^{k-i})
\end{aligned}
$$

Lower bounds

The iteration of gradient descent:

$$
\begin{aligned}
x^{k+1} &= x^k - \alpha^k \nabla f(x^k) \\
&= x^{k-1} - \alpha^{k-1} \nabla f(x^{k-1}) - \alpha^k \nabla f(x^k) \\
&\;\vdots \\
&= x^0 - \sum_{i=0}^{k} \alpha^{k-i} \nabla f(x^{k-i})
\end{aligned}
$$

Consider a family of first-order methods, where

$$
x^{k+1} \in x^0 + \text{span}\left\{\nabla f(x^0), \nabla f(x^1), \ldots, \nabla f(x^k)\right\} \quad (1)
$$

Lower bounds

The iteration of gradient descent:

$$
\begin{aligned}
x^{k+1} &= x^k - \alpha^k \nabla f(x^k) \\
&= x^{k-1} - \alpha^{k-1} \nabla f(x^{k-1}) - \alpha^k \nabla f(x^k) \\
&\ \vdots \\
&= x^0 - \sum_{i=0}^{k} \alpha^{k-i} \nabla f(x^{k-i})
\end{aligned}
$$

Consider a family of first-order methods, where

$$
x^{k+1} \in x^0 + \text{span}\left\{\nabla f(x^0), \nabla f(x^1), \ldots, \nabla f(x^k)\right\} \quad (1)
$$

Non-smooth convex case

There exists a function f that is M-Lipschitz and convex such that any first-order method of the form 1 satisfies

$$
\min_{i \in [1,k]} f(x^i) - f^* \geq \frac{M \|x^0 - x^*\|_2}{2(1 + \sqrt{k})}
$$

Lower bounds

The iteration of gradient descent:

$$
\begin{aligned}
x^{k+1} &= x^k - \alpha^k \nabla f(x^k) \\
&= x^{k-1} - \alpha^{k-1} \nabla f(x^{k-1}) - \alpha^k \nabla f(x^k) \\
&\vdots \\
&= x^0 - \sum_{i=0}^{k} \alpha^{k-i} \nabla f(x^{k-i})
\end{aligned}
$$

Consider a family of first-order methods, where

$$
x^{k+1} \in x^0 + \text{span}\left\{\nabla f(x^0), \nabla f(x^1), \ldots, \nabla f(x^k)\right\} \quad (1)
$$

Non-smooth convex case

There exists a function f that is M-Lipschitz and convex such that any first-order method of the form 1 satisfies

$$
\min_{i \in [1,k]} f(x^i) - f^* \geq \frac{M \|x^0 - x^*\|_2}{2(1+\sqrt{k})}
$$

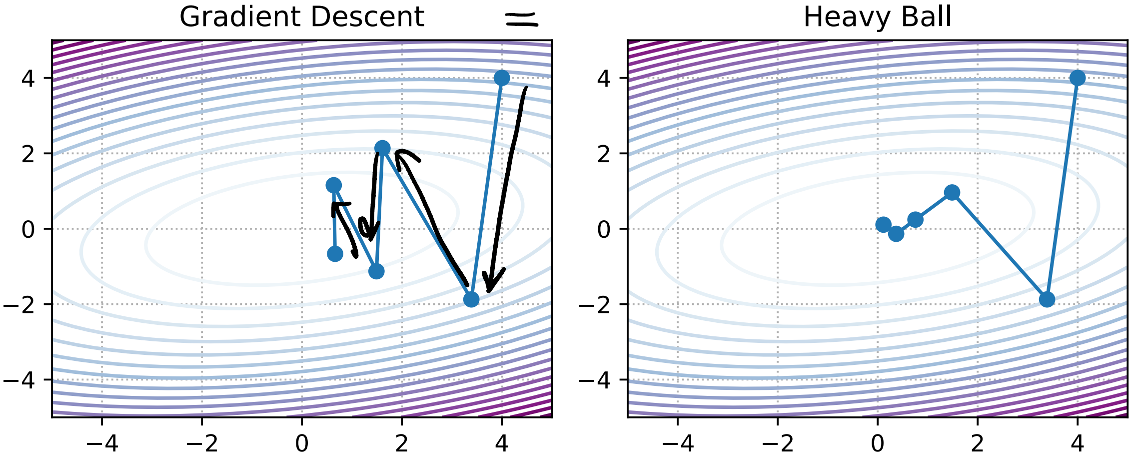
Smooth and convex case

There exists a function f that is L-smooth and convex such that any first-order method of the form 1 satisfies

$$
\min_{i \in [1,k]} f(x^i) - f^* \geq \frac{3L \|x^0 - x^*\|_2^2}{32(1+k)^2}
$$

Oscillations and acceleration

$$x_{k+1} = x_k - \alpha_k \nabla f(x_k) + \beta(x_k - x_{k-1})$$



$$f(x) = \frac{1}{2} x^T A x$$

$$\nabla f = A x$$

$$x_{k+1} = x_k - \alpha \cdot A x_k + \beta \left(x_k - x_{k-1} \right) =$$

$$x_k - x_{k-1} = -\alpha A x_{k-1} + \beta \left(x_{k-1} - x_{k-2} \right)$$

$$= x_k - \alpha A x_k + \beta \left(-\alpha A x_{k-1} + \beta \left(x_{k-1} - x_{k-2} \right) \right)$$

Coordinate shift

Consider the following quadratic optimization problem:

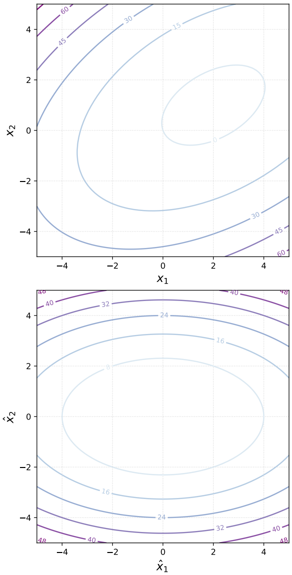
$$
\min_{x \in \mathbb{R}^d} f(x) = \min_{x \in \mathbb{R}^d} \frac{1}{2} x^\top A x - b^\top x + c, \text{ where } A \in \mathbb{S}^d_{++}.
$$

Coordinate shift

Consider the following quadratic optimization problem:

$$
\min_{x \in \mathbb{R}^d} f(x) = \min_{x \in \mathbb{R}^d} \frac{1}{2} x^\top A x - b^\top x + c, \text{ where } A \in \mathbb{S}^d_{++}.
$$

- Firstly, without loss of generality we can set $c = 0$, which will or affect optimization process.



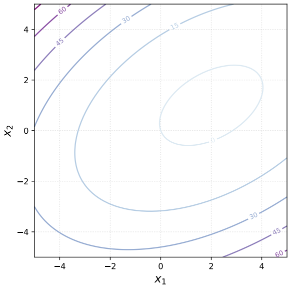
Coordinate shift

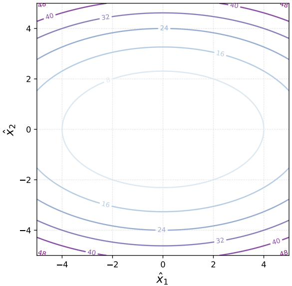
Consider the following quadratic optimization problem:

$$
\min_{x \in \mathbb{R}^d} f(x) = \min_{x \in \mathbb{R}^d} \frac{1}{2} x^\top A x - b^\top x + c, \text{ where } A \in \mathbb{S}^d_{++}.
$$

- Firstly, without loss of generality we can set $c = 0$, which will or affect optimization process.
- Secondly, we have a spectral decomposition of the matrix A:

$$
A = Q \Lambda Q^T
$$





Coordinate shift

Consider the following quadratic optimization problem:

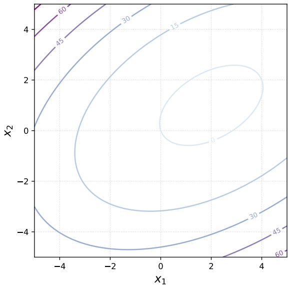
$$
\min_{x \in \mathbb{R}^d} f(x) = \min_{x \in \mathbb{R}^d} \frac{1}{2} x^\top A x - b^\top x + c, \text{ where } A \in \mathbb{S}^d_{++}.
$$

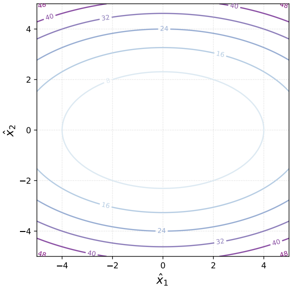
- Firstly, without loss of generality we can set $c = 0$, which will or affect optimization process.
- Secondly, we have a spectral decomposition of the matrix A:

$$
A = Q \Lambda Q^T
$$

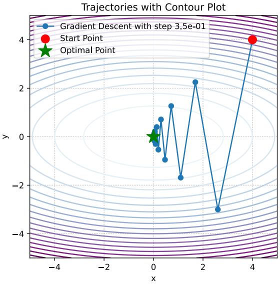
- Let's show, that we can switch coordinates in order to make an analysis a little bit easier. Let $\hat{x} = Q^T(x - x^*)$, where x^* is the minimum point of initial function, defined by $Ax^* = b$. At the same time $x = Q\hat{x} + x^*$.

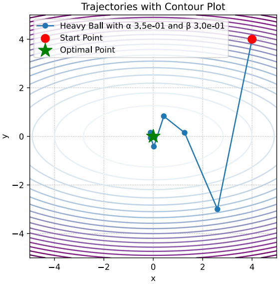
$$
\begin{aligned}
f(\hat{x}) &= \frac{1}{2}(Q\hat{x} + x^*)^\top A (Q\hat{x} + x^*) - b^\top (Q\hat{x} + x^*) \\
&= \frac{1}{2}\hat{x}^T Q^T A Q \hat{x} + (x^*)^T A Q \hat{x} + \frac{1}{2}(x^*)^T A (x^*)^T - b^T Q \hat{x} - b^T x^* \\
&= \frac{1}{2}\hat{x}^T \Lambda \hat{x}
\end{aligned}
$$





Polyak Heavy ball method

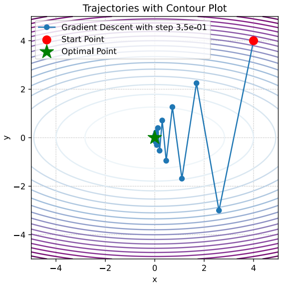


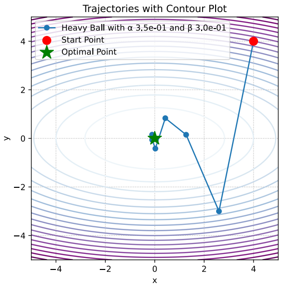


Let's introduce the idea of momentum, proposed by Polyak in 1964. Recall that the momentum update is

$$\boxed{x^{k+1} = x^k - \alpha \nabla f(x^k) + \beta (x^k - x_{k-1}).}$$

Polyak Heavy ball method





Let's introduce the idea of momentum, proposed by Polyak in 1964. Recall that the momentum update is

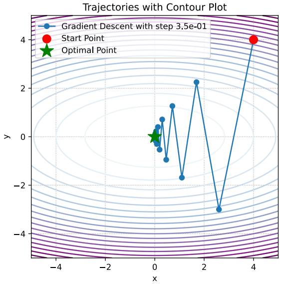
$\nabla f = \Lambda x$

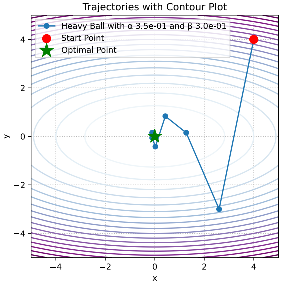
$$x^{k+1} = x^k - \alpha \nabla f(x^k) + \beta(x^k - x_{k-1}).$$

Which is in our (quadratics) case is

$$\hat{x}_{k+1} = \hat{x}_k - \alpha \Lambda \hat{x}_k + \beta(\hat{x}_k - \hat{x}_{k-1}) = (I - \alpha\Lambda + \beta I)\hat{x}_k - \beta \hat{x}_{k-1}$$

Polyak Heavy ball method





Let's introduce the idea of momentum, proposed by Polyak in 1964. Recall that the momentum update is

$$x^{k+1} = x^k - \alpha \nabla f(x^k) + \beta (x^k - x_{k-1}).$$

Which is in our (quadratics) case is

$$\hat{x}_{k+1} = \hat{x}_k - \alpha \Lambda \hat{x}_k + \beta(\hat{x}_k - \hat{x}_{k-1}) = (I - \alpha\Lambda + \beta I)\hat{x}_k - \beta \hat{x}_{k-1}$$

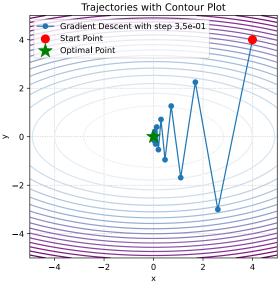
This can be rewritten as follows

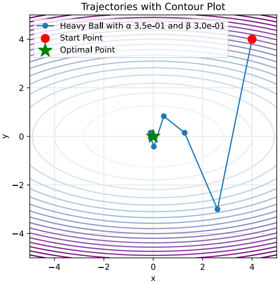
$$\hat{x}_{k+1} = (I - \alpha\Lambda + \beta I)\hat{x}_k - \beta \hat{x}_{k-1},$$
$$\hat{x}_k = \hat{x}_k.$$

$$z_{k+1} = M z_k$$

$$z_k = \begin{pmatrix} x_{k+1} \\ x_k \end{pmatrix}$$

Polyak Heavy ball method





Let's introduce the idea of momentum, proposed by Polyak in 1964. Recall that the momentum update is

$$x^{k+1} = x^k - \alpha \nabla f(x^k) + \beta(x^k - x_{k-1}).$$

Which is in our (quadratics) case is

$$\hat{x}_{k+1} = \hat{x}_k - \alpha \Lambda \hat{x}_k + \beta(\hat{x}_k - \hat{x}_{k-1}) = (I - \alpha\Lambda + \beta I)\hat{x}_k - \beta \hat{x}_{k-1}$$

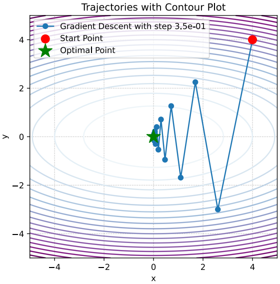
This can be rewritten as follows

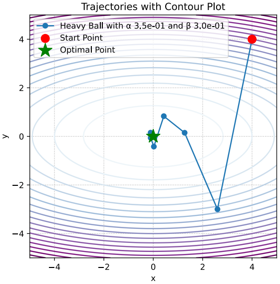
$$\begin{cases} \hat{x}_{k+1} = (I - \alpha\Lambda + \beta I)\hat{x}_k - \beta \hat{x}_{k-1}, \\ \hat{x}_k = \hat{x}_k. \quad + 0\, x_{k-1} \end{cases}$$

2n

Let's use the following notation $\boxed{\hat{z}_k = \begin{bmatrix} \hat{x}_{k+1} \\ \hat{x}_k \end{bmatrix}}$. Therefore $\boxed{\hat{z}_{k+1} = M \hat{z}_k}$, where the iteration matrix M is:

Polyak Heavy ball method





Let's introduce the idea of momentum, proposed by Polyak in 1964. Recall that the momentum update is

$$x^{k+1} = x^k - \alpha \nabla f(x^k) + \beta(x^k - x_{k-1}).$$

Which is in our (quadratics) case is

$$\hat{x}_{k+1} = \hat{x}_k - \alpha \Lambda \hat{x}_k + \beta(\hat{x}_k - \hat{x}_{k-1}) = (I - \alpha\Lambda + \beta I)\hat{x}_k - \beta \hat{x}_{k-1}$$

This can be rewritten as follows

$z_k = \begin{matrix} x_{k+1} \\ x_k \end{matrix}$

$$\begin{aligned} \hat{x}_{k+1} &= (I - \alpha\Lambda + \beta I)\hat{x}_k - \beta \hat{x}_{k-1}, \\ \hat{x}_k &= \hat{x}_k. \end{aligned}$$

$z_{k-1} = \begin{matrix} x_k \\ x_{k-1} \end{matrix}$

Let's use the following notation $\hat{z}_k = \begin{bmatrix} \hat{x}_{k+1} \\ \hat{x}_k \end{bmatrix}$. Therefore $\hat{z}_{k+1} = M\hat{z}_k$, where the iteration matrix M is:

2d

$$M = \begin{bmatrix} I - \alpha\Lambda + \beta I & -\beta I \\ I & 0_d \end{bmatrix}.$$

2d

Reduction to a scalar case

Note, that M is $2d \times 2d$ matrix with 4 block-diagonal matrices of size $d \times d$ inside. It means, that we can rearrange the order of coordinates to make M block-diagonal in the following form. Note that in the equation below, the matrix M denotes the same as in the notation above, except for the described permutation of rows and columns. We use this slight abuse of notation for the sake of clarity.

$$z_{k+1} = M z_k$$

$$\rho(M) < 1 \Leftarrow \text{CX-Tb}$$

$$\rho(M) = \max |\lambda(M)|$$

Reduction to a scalar case

Note, that M is $2d \times 2d$ matrix with 4 block-diagonal matrices of size $d \times d$ inside. It means, that we can rearrange the order of coordinates to make M block-diagonal in the following form. Note that in the equation below, the matrix M denotes the same as in the notation above, except for the described permutation of rows and columns. We use this slight abuse of notation for the sake of clarity.

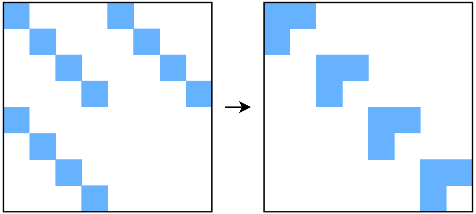


Figure 1: Illustration of matrix M rearrangement

$$
\begin{bmatrix} \hat{x}_k^{(1)} \\ \vdots \\ \hat{x}_k^{(d)} \\ \hat{x}_{k-1}^{(1)} \\ \vdots \\ \hat{x}_{k-1}^{(d)} \end{bmatrix} \to \begin{bmatrix} \hat{x}_k^{(1)} \\ \hat{x}_{k-1}^{(1)} \\ \vdots \\ \hat{x}_k^{(d)} \\ \hat{x}_{k-1}^{(d)} \end{bmatrix} \quad M = \begin{bmatrix} M_1 & & & \\ & M_2 & & \\ & & \ldots & \\ & & & M_d \end{bmatrix}
$$

where $\hat{x}_k^{(i)}$ is i-th coordinate of vector $\hat{x}_k \in \mathbb{R}^d$ and M_i stands for 2×2 matrix. This rearrangement allows us to study the dynamics of the method independently for each dimension. One may observe, that the asymptotic convergence rate of the $2d$-dimensional vector sequence of $\hat{z}_k$ is defined by the worst convergence rate among its block of coordinates. Thus, it is enough to study the optimization in a one-dimensional case.

Reduction to a scalar case

For i-th coordinate with λ_i as an i-th eigenvalue of matrix W we have:

$$M_i = \begin{bmatrix} 1 - \alpha\lambda_i + \beta & -\beta \\ 1 & 0 \end{bmatrix}.$$

$$\begin{pmatrix} x^{k+1}_{(i)} \\ x^{k}_{(i)} \end{pmatrix} = M_i \begin{pmatrix} x^{k}_{(i)} \\ x^{k-1}_{(i)} \end{pmatrix}$$

Reduction to a scalar case

For i-th coordinate with λ_i as an i-th eigenvalue of matrix W we have:

$$M_i = \begin{bmatrix} 1 - \alpha\lambda_i + \beta & -\beta \\ 1 & 0 \end{bmatrix}.$$

The method will be convergent if $\rho(M) < 1$, and the optimal parameters can be computed by optimizing the spectral radius

$$\boxed{\alpha^*, \beta^* = \arg\min_{\alpha,\beta} \max_{\lambda\in[\mu,L]} \rho(M)} \quad \alpha^* = \frac{4}{(\sqrt{L}+\sqrt{\mu})^2}; \quad \beta^* = \left(\frac{\sqrt{L}-\sqrt{\mu}}{\sqrt{L}+\sqrt{\mu}}\right)^2.$$

Reduction to a scalar case

For i-th coordinate with λ_i as an i-th eigenvalue of matrix W we have:

$$
M_i = \begin{bmatrix} 1 - \alpha\lambda_i + \beta & -\beta \\ 1 & 0 \end{bmatrix}.
$$

The method will be convergent if $\rho(M) < 1$, and the optimal parameters can be computed by optimizing the spectral radius

$$
\alpha^*, \beta^* = \arg\min_{\alpha,\beta} \max_{\cancel{\lambda \in [\mu, L]}\, i} \rho(M_i) \quad \alpha^* = \frac{4}{(\sqrt{L} + \sqrt{\mu})^2}; \quad \beta^* = \left(\frac{\sqrt{L} - \sqrt{\mu}}{\sqrt{L} + \sqrt{\mu}}\right)^2.
$$

It can be shown, that for such parameters the matrix M has complex eigenvalues, which forms a conjugate pair, so the distance to the optimum (in this case, $\|z_k\|$), generally, will not go to zero monotonically.

Heavy ball quadratic convergence

We can explicitly calculate the eigenvalues of M_i:

$$\lambda_1^M, \lambda_2^M = \lambda\left(\begin{bmatrix} 1-\alpha\lambda_i+\beta & -\beta \\ 1 & 0 \end{bmatrix}\right) = \frac{1+\beta-\alpha\lambda_i \pm \sqrt{(1+\beta-\alpha\lambda_i)^2-4\beta}}{2}.$$

Heavy ball quadratic convergence

We can explicitly calculate the eigenvalues of M_i:

$$\lambda^M_1, \lambda^M_2 = \lambda \left(\begin{bmatrix} 1 - \alpha\lambda_i + \beta & -\beta \\ 1 & 0 \end{bmatrix} \right) = \frac{1 + \beta - \alpha\lambda_i \pm \sqrt{(1 + \beta - \alpha\lambda_i)^2 - 4\beta}}{2}.$$

When α and β are optimal (α^*, β^*), the eigenvalues are complex-conjugated pair $(1 + \beta - \alpha\lambda_i)^2 - 4\beta \leq 0$, i.e. $\beta \geq (1 - \sqrt{\alpha\lambda_i})^2$.

Heavy ball quadratic convergence

We can explicitly calculate the eigenvalues of M_i:

$$\lambda^M_1, \lambda^M_2 = \lambda \left(\begin{bmatrix} 1 - \alpha\lambda_i + \beta & -\beta \\ 1 & 0 \end{bmatrix} \right) = \frac{1 + \beta - \alpha\lambda_i \pm \sqrt{(1 + \beta - \alpha\lambda_i)^2 - 4\beta}}{2}.$$

When α and β are optimal (α^*, β^*), the eigenvalues are complex-conjugated pair $(1 + \beta - \alpha\lambda_i)^2 - 4\beta \leq 0$, i.e. $\beta \geq (1 - \sqrt{\alpha\lambda_i})^2$.

$$\text{Re}(\lambda^M_1) = \frac{L + \mu - 2\lambda_i}{(\sqrt{L} + \sqrt{\mu})^2}; \quad \text{Im}(\lambda^M_1) = \frac{\pm 2\sqrt{(L - \lambda_i)(\lambda_i - \mu)}}{(\sqrt{L} + \sqrt{\mu})^2}; \quad |\lambda^M_1| = \frac{L - \mu}{(\sqrt{L} + \sqrt{\mu})^2}.$$

Heavy ball quadratic convergence

We can explicitly calculate the eigenvalues of M_i:

$$\lambda^M_1, \lambda^M_2 = \lambda \left(\begin{bmatrix} 1 - \alpha\lambda_i + \beta & -\beta \\ 1 & 0 \end{bmatrix} \right) = \frac{1 + \beta - \alpha\lambda_i \pm \sqrt{(1 + \beta - \alpha\lambda_i)^2 - 4\beta}}{2}.$$

When α and β are optimal (α^*, β^*), the eigenvalues are complex-conjugated pair $(1 + \beta - \alpha\lambda_i)^2 - 4\beta \leq 0$, i.e. $\beta \geq (1 - \sqrt{\alpha\lambda_i})^2$.

$$\text{Re}(\lambda^M_1) = \frac{L + \mu - 2\lambda_i}{(\sqrt{L} + \sqrt{\mu})^2}; \quad \text{Im}(\lambda^M_1) = \frac{\pm 2\sqrt{(L - \lambda_i)(\lambda_i - \mu)}}{(\sqrt{L} + \sqrt{\mu})^2}; \quad |\lambda^M_1| = \frac{L - \mu}{(\sqrt{L} + \sqrt{\mu})^2}.$$

And the convergence rate does not depend on the stepsize and equals to $\sqrt{\beta^*}$.

Heavy Ball quadratics convergence

$L = \lambda_{max}(\nabla^2 f(x))$ $\qquad \varkappa = \frac{L}{\mu} \geq 1$

$\mu = \lambda_{min}(\nabla^2 f(x)) > 0$

Theorem

Assume that f is quadratic μ-strongly convex L-smooth quadratics, then Heavy Ball method with parameters

$$\alpha = \frac{4}{(\sqrt{L} + \sqrt{\mu})^2}, \beta = \frac{\sqrt{L} - \sqrt{\mu}}{\sqrt{L} + \sqrt{\mu}}$$

converges linearly:

$$\|x_k - x^*\|_2 \leq \left(\frac{\sqrt{\kappa} - 1}{\sqrt{\kappa} + 1} \right) \|x_0 - x^*\|$$

ускоренный rate

Heavy Ball Global Convergence [3]

$O\left(\frac{1}{T}\right)$

Theorem

Assume that f is smooth and convex and that

$$
\beta \in [0,1), \quad \alpha \in \left(0, \frac{2(1-\beta)}{L}\right).
$$

Then, the sequence $\{x_k\}$ generated by Heavy-ball iteration satisfies

$$
f(\overline{x}_T) - f^\star \leq \begin{cases} \frac{\|x_0 - x^\star\|^2}{2(T+1)}\left(\frac{L\beta}{1-\beta} + \frac{1-\beta}{\alpha}\right), & \text{if } \alpha \in \left(0, \dfrac{1-\beta}{L}\right], \\ \frac{\|x_0 - x^\star\|^2}{2(T+1)(2(1-\beta)-\alpha L)}\left(L\beta + \frac{(1-\beta)^2}{\alpha}\right), & \text{if } \alpha \in \left[\dfrac{1-\beta}{L}, \dfrac{2(1-\beta)}{L}\right), \end{cases}
$$

where $\overline{x}_T$ is the Cesaro average of the iterates, i.e.,

$$
\overline{x}_T = \frac{1}{T+1}\sum_{k=0}^{T} x_k.
$$

[3] Global convergence of the Heavy-ball method for convex optimization, Euhanna Ghadimi et.al.

Heavy Ball Global Convergence [4]

Theorem

Assume that f is smooth and strongly convex and that

$$\alpha \in (0, \frac{2}{L}), \quad 0 \leq \beta < \frac{1}{2}\left(\frac{\mu\alpha}{2} + \sqrt{\frac{\mu^2\alpha^2}{4} + 4(1 - \frac{\alpha L}{2})} \right).$$

where $\alpha_0 \in (0, 1/L]$. Then, the sequence $\{x_k\}$ generated by Heavy-ball iteration converges linearly to a unique optimizer $x^\star$. In particular,

$$f(x_k) - f^\star \leq q^k (f(x_0) - f^\star),$$

where $q \in [0, 1)$.

[4] Global convergence of the Heavy-ball method for convex optimization, Euhanna Ghadimi et.al.

Heavy ball method summary

- Ensures accelerated convergence for strongly convex quadratic problems

Heavy ball method summary

- Ensures accelerated convergence for strongly convex quadratic problems
- Local accelerated convergence was proved in the original paper.

Heavy ball method summary

* Ensures accelerated convergence for strongly convex quadratic problems
* Local accelerated convergence was proved in the original paper.
* Recently was proved, that there is no global accelerated convergence for the method.

Heavy ball method summary

* Ensures accelerated convergence for strongly convex quadratic problems
* Local accelerated convergence was proved in the original paper.
* Recently was proved, that there is no global accelerated convergence for the method.
* Method was not extremely popular until the ML boom

Heavy ball method summary

- Ensures accelerated convergence for strongly convex quadratic problems
- Local accelerated convergence was proved in the original paper.
- Recently was proved, that there is no global accelerated convergence for the method.
- Method was not extremely popular until the ML boom
- Nowadays, it is de-facto standard for practical acceleration of gradient methods, even for the non-convex problems (neural network training)

The concept of Nesterov Accelerated Gradient method

1983

$$\boxed{x_{k+1} = x_k - \alpha \nabla f(x_k)}$$

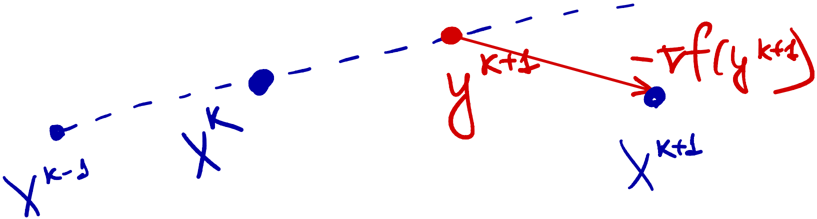
Коши GD

18...47

$$\boxed{x_{k+1} = x_k - \alpha \nabla f(x_k) + \beta(x_k - x_{k-1})}$$

1964 HB

$$\begin{cases} y_{k+1} = x_k + \beta(x_k - x_{k-1}) \\ x_{k+1} = y_{k+1} - \alpha \nabla f(y_{k+1}) \end{cases}$$



The concept of Nesterov Accelerated Gradient method

$$x_{k+1} = x_k - \alpha \nabla f(x_k) \qquad x_{k+1} = x_k - \alpha \nabla f(x_k) + \beta(x_k - x_{k-1}) \qquad \begin{cases} y_{k+1} = x_k + \beta(x_k - x_{k-1}) \\ x_{k+1} = y_{k+1} - \alpha \nabla f(y_{k+1}) \end{cases}$$

Let's define the following notation

$$x^+ = x - \alpha \nabla f(x) \qquad \text{Gradient step}$$
$$d_k = \beta_k (x_k - x_{k-1}) \qquad \text{Momentum term}$$

Then we can write down:

$$x_{k+1} = x_k^+ \qquad \text{Gradient Descent}$$
$$x_{k+1} = x_k^+ + d_k \qquad \text{Heavy Ball}$$
$$x_{k+1} = (x_k + d_k)^+ \qquad \text{Nesterov accelerated gradient}$$

NAG convergence for quadratics

General case convergence

Theorem

Let $f : \mathbb{R}^n \to \mathbb{R}$ is convex and L-smooth. The Nesterov Accelerated Gradient Descent (NAG) algorithm is designed to solve the minimization problem starting with an initial point $x_0 = y_0 \in \mathbb{R}^n$ and $\lambda_0 = 0$. The algorithm iterates the following steps:

$$
\begin{aligned}
&\textbf{Gradient update:} && y_{k+1} = x_k - \frac{1}{L}\nabla f(x_k) \\
&\textbf{Extrapolation:} && x_{k+1} = (1-\gamma_k)y_{k+1} + \gamma_k y_k \\
&\textbf{Extrapolation weight:} && \lambda_{k+1} = \frac{1 + \sqrt{1 + 4\lambda_k^2}}{2} \\
&\textbf{Extrapolation weight:} && \gamma_k = \frac{1-\lambda_k}{\lambda_{k+1}}
\end{aligned}
$$

The sequences $\{f(y_k)\}_{k\in\mathbb{N}}$ produced by the algorithm will converge to the optimal value f^* at the rate of $\mathcal{O}\left(\frac{1}{k^2}\right)$, specifically:

$$
f(y_k) - f^* \leq \frac{2L\|x_0 - x^*\|^2}{k^2}
$$

General case convergence

Theorem

Let $f : \mathbb{R}^n \to \mathbb{R}$ is μ-strongly convex and L-smooth. The Nesterov Accelerated Gradient Descent (NAG) algorithm is designed to solve the minimization problem starting with an initial point $x_0 = y_0 \in \mathbb{R}^n$ and $\lambda_0 = 0$. The algorithm iterates the following steps:

Gradient update: $$\boxed{y_{k+1} = x_k - \frac{1}{L}\nabla f(x_k)}$$

Extrapolation: $$\boxed{x_{k+1} = (1-\gamma_k) y_{k+1} + \gamma_k y_k}$$

Extrapolation weight: $$\gamma_k = \frac{\sqrt{L} - \sqrt{\mu}}{\sqrt{L} + \sqrt{\mu}}$$

The sequences $\{f(y_k)\}_{k \in \mathbb{N}}$ produced by the algorithm will converge to the optimal value f^* linearly:

$$f(y_k) - f^* \leq \frac{\mu + L}{2} \|x_0 - x^*\|_2^2 \exp\left(-\frac{k}{\sqrt{\kappa}}\right)$$