IE510 Applied Nonlinear Programming

Lecture 3: Heavy Ball Method and Momentum

Ruoyu Sun

Feb 6, 2018

Last Time: Convergence Rate

- **Proposition 3b**: Suppose f is a strongly convex function, and

$$\mu I \preceq \nabla^2 f(\mathbf{x}) \preceq LI, \quad \forall \mathbf{x} \in \mathbb{R}^n.$$

Consider

$$\min_{\mathbf{x}\in\mathbb{R}^n} f(\mathbf{x}).$$

Suppose GD with stepsize $1/L$ generates a sequence $\mathbf{x}^r$, then

$$\frac{e^r}{e^0} = \frac{f(\mathbf{x}^r) - f^*}{f(\mathbf{x}^0) - f^*} \leq \left(1 - \frac{1}{\kappa}\right)^r.$$

where the condition number $\kappa = \frac{L}{\sigma}$.

- Remark 1: $O(\kappa \log(1/\epsilon))$ iterations to achieve relative error ϵ
- **Practical Lesson**: Normalizing data might reduce condition number, which makes GD faster.

Last Time: Convergence Rate

- **Proposition 3b**: Suppose f is a strongly convex function, and

$$\mu I \preceq \nabla^2 f(\mathbf{x}) \preceq LI, \quad \forall \mathbf{x} \in \mathbb{R}^n.$$

Consider

$$\min_{\mathbf{x}\in\mathbb{R}^n} f(\mathbf{x}).$$

Suppose GD with stepsize $1/L$ generates a sequence $\mathbf{x}^r$, then

$$\frac{e^r}{e^0} = \frac{f(\mathbf{x}^r) - f^*}{f(\mathbf{x}^0) - f^*} \leq \left(1 - \frac{1}{\kappa}\right)^r.$$

where the condition number $\kappa = \frac{L}{\sigma}$.

- Remark 1: $O(\kappa \log(1/\epsilon))$ iterations to achieve relative error ϵ
- **Practical Lesson**: Normalizing data might reduce condition number, which makes GD faster.

Questions for Last Week

- **Q1: First**, how to ensure your algorithm is "convergent"?

  –Armijo rule or GD with $< 2/L$ stepsize

  –sequence not diverging to ∞ ($\|x^r\| \nrightarrow \infty$)

- **Q2**: You work for an IT company in advertisement prediction.

  You observe error $1.8, 1.7, \ldots, 0.94, 0.93, 0.92, 0.915, 0.910, 0.908, 0.906$ after 1000 iterations, 5 hours training.

  What can you say? Does it converge already?

  Hard to say. Maybe estimating condition # helps a bit... (but need to know hw 2 problem 2)

- **Two possibilities**:
  - Already converges – you should try different _____ then
  - Not yet converges – you should try different _____ then

Questions for Last Week

- **Q1: First**, how to ensure your algorithm is "convergent"?

  –Armijo rule or GD with $< 2/L$ stepsize

  –sequence not diverging to ∞ ($\|x^r\| \nrightarrow \infty$)

- **Q2**: You work for an IT company in advertisement prediction.

  You observe error $1.8, 1.7, \ldots, 0.94, 0.93, 0.92, 0.915, 0.910, 0.908, 0.906$ after 1000 iterations, 5 hours training.

  What can you say? Does it converge already?

  Hard to say. Maybe estimating condition # helps a bit... (but need to know hw 2 problem 2)

- **Two possibilities**:
  - Already converges – you should try different _____ then
  - Not yet converges – you should try different _____ then

Questions for Last Week

- **Q1: First**, how to ensure your algorithm is "convergent"?

  –Armijo rule or GD with $< 2/L$ stepsize

  –sequence not diverging to ∞ ($\|x^r\| \nrightarrow \infty$)

- **Q2**: You work for an IT company in advertisement prediction.

  You observe error $1.8, 1.7, \ldots, 0.94, 0.93, 0.92, 0.915, 0.910, 0.908, 0.906$ after 1000 iterations, 5 hours training.

  What can you say? Does it converge already?

  Hard to say. Maybe estimating condition # helps a bit... (but need to know hw 2 problem 2)

- **Two possibilities**:
  - Already converges – you should try different _____ then
  - Not yet converges – you should try different _____ then

Questions for Last Week

- **Q1: First**, how to ensure your algorithm is "convergent"?

  –Armijo rule or GD with $< 2/L$ stepsize

  –sequence not diverging to ∞ ($\|x^r\| \nrightarrow \infty$)

- **Q2**: You work for an IT company in advertisement prediction.

  You observe error $1.8, 1.7, \ldots, 0.94, 0.93, 0.92, 0.915, 0.910, 0.908, 0.906$ after 1000 iterations, 5 hours training.

  0.002 0.002,

  What can you say? Does it converge already?

  Hard to say. Maybe estimating condition # helps a bit... (but need to know hw 2 problem 2)

- **Two possibilities**:
  - Already converges – you should try different model then
  - Not yet converges – you should try different algorithm then

  0.805, in 1 day.

Discussion of Question 2.

Group 1: (1) Hard to tell; the analysis is for convex case, but the practical problem may be non-convex.

(2) Can try different stepsize rules.

Group 2: Depending on application; if the error is small enough, say, 1% error, then I don't care whether converge.

Response to Group 2:

Yes, if error is small enough, no need to check convergence.

But in this problem (predicting click rate of advertisement), the error 0.905 is still quite large. So need to know whether converge or not.

Today

- Announcement: Homework 2 due next Tuesday. Change to Thursday, 02/15.

  Simplest experiment + analysis. Help you understand what the computer is doing!

- Today: GD with momentum; i.e., heavy ball method

- After today's course, you will be able to

  - pick the parameters of GD with momentum
  - explain why momentum is useful

Outline

Next Few Weeks: Faster Algorithm

Lesson: For huge problems, "slow" $\approx$ "not converging"

Convergence speed is as important as convergence itself, if not more.

Three classes of faster algorithms (besides preconditioning)

- Using momentum : today
- Using spectral info
- Decomposition (into small problems)

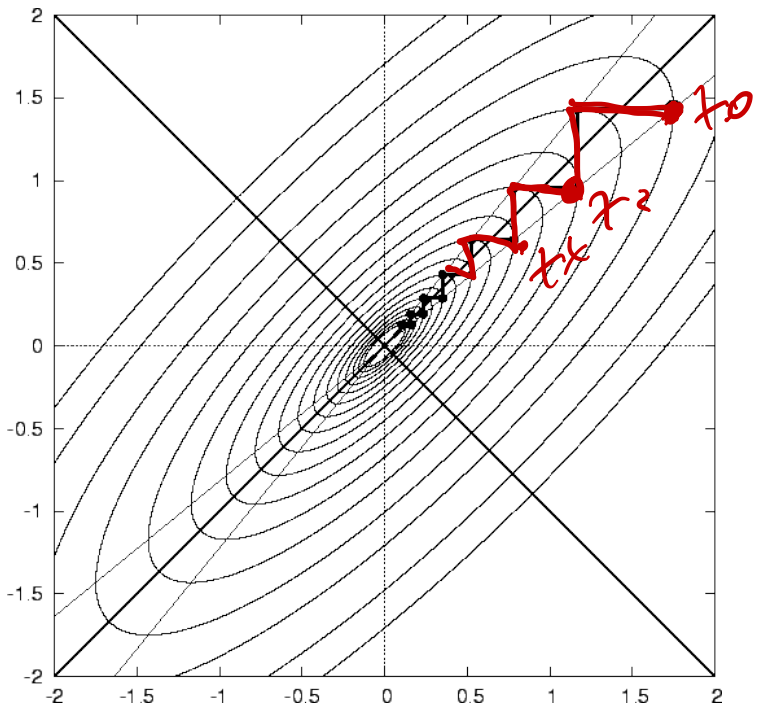
20+ different algorithms/variants. We'll cover some of them.

The more (algorithms you know), the better !

Sin of Greedy

Recall GD: a local greedy method

"*Those who cannot remember the past are condemned to repeat it.*"
–Santayana

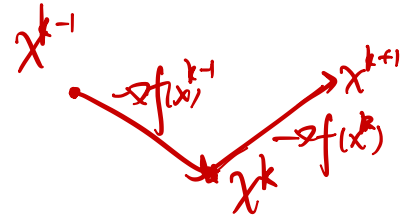


For GD, can we incorporate history to improve?

Add History Info to GD

Gradient descent is going too fast on the new direction.

$$\text{GD}: x^{k+1} = x^k - \alpha \nabla f(x^k).$$



Use the information of the old direction.

GD with momentum (**heavy ball** method): [Polyak'1974]

$$\text{Heavy ball}: x^{k+1} = x^k - \alpha \nabla f(x^k) + \beta (x^k - x^{k-1})$$

momentum

$-\nabla f(x^{k-1})$. candidate.

Fixed point verification: $x^\infty = x^\infty - \alpha \nabla f(x^\infty) + \beta(x^\infty - x^\infty) \Rightarrow \nabla f(x^\infty) = 0$.

Remark (don't forget history): Lots of similar methods before Polyak'1974 were proposed.

Constant α, β for linear algebra were studied by Frankel'1950 (call it 2nd order Richardson method).

Momentum is Popular

Heavy ball method is not commonly covered in traditional optimization courses.

but popularized by ML community recently, under the name “GD with momentum”

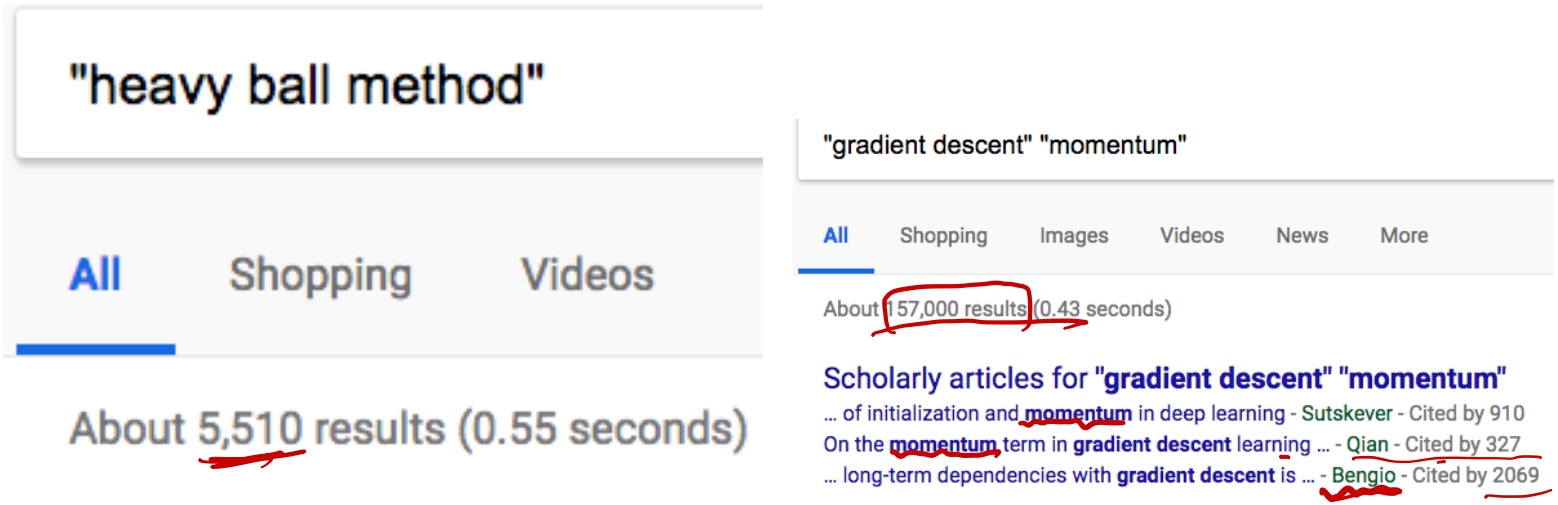


Figure: Left: Heavy ball; Right: Momentum

Momentum is Popular

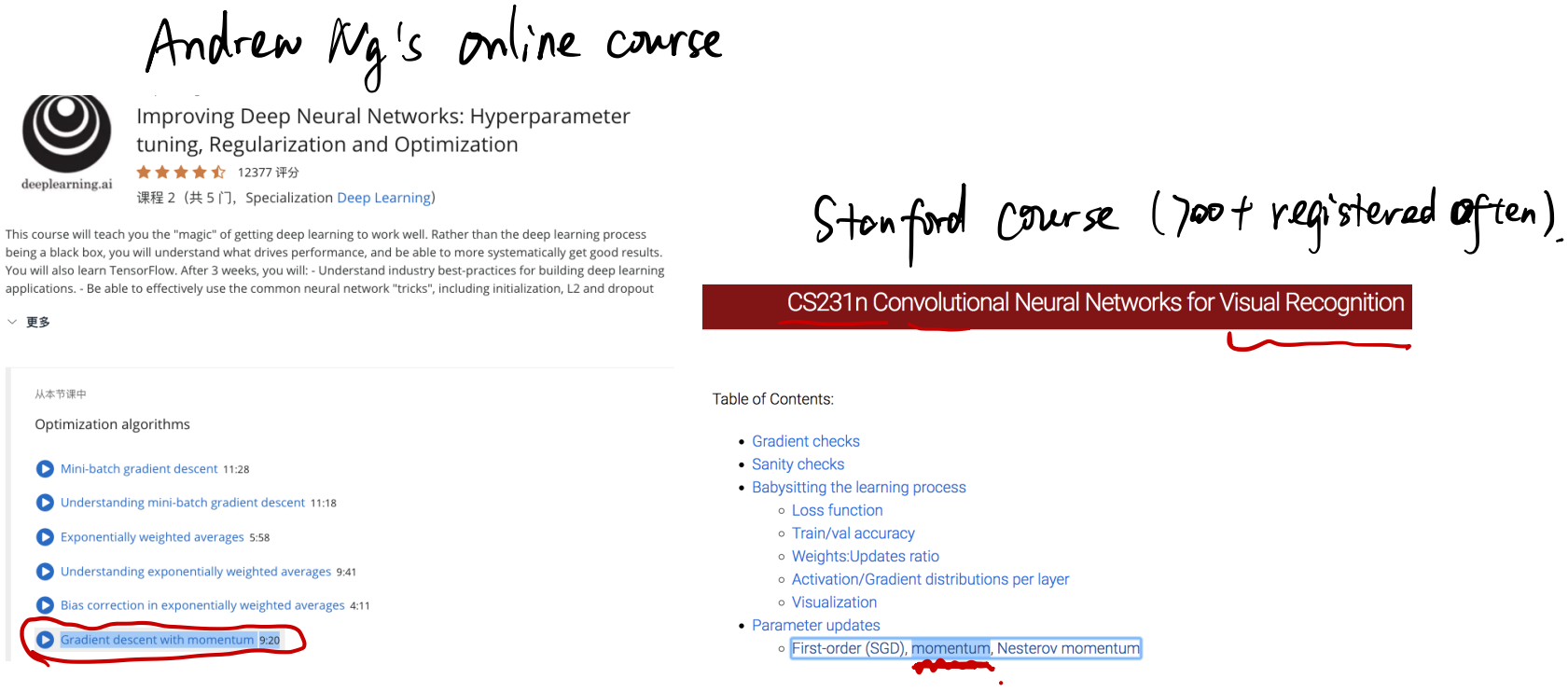


Figure: Snapshots of Online Resources on Momentum

Momentum is Popular (cont'd)

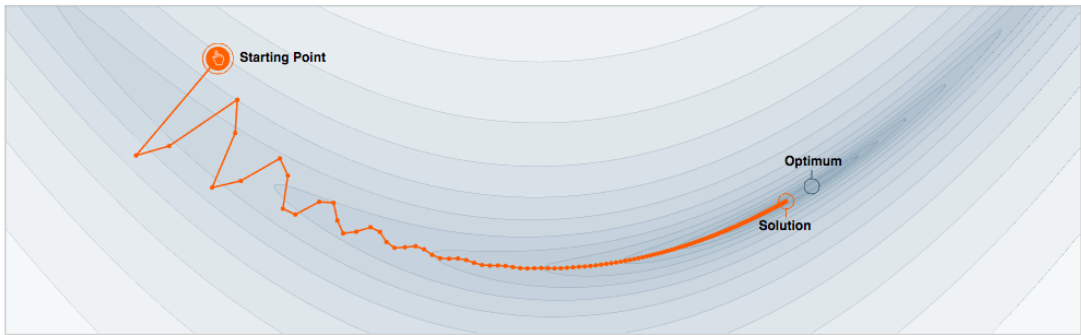
Conclusion

In this post we looked at the optimization algorithms for neural nets beyond SGD. We looked at two classes of algorithms: momentum based and adaptive learning rate methods.

We also implement all of those methods in Python and Numpy with the use case of our neural nets stated in the last post.

Most of those methods above are currently implemented in the popular Deep Learning

Why Momentum Really Works



Step-size α = 0.02

0 0.003 0.006

Momentum β = 0.99

0.00 0.500 0.990

We often think of Momentum as a means of dampening oscillations and speeding up the iterations, leading to faster convergence. But it has other interesting behavior. It allows a larger range of step-sizes to be used, and creates its own oscillations. What is going on?

GABRIEL GOH
UC Davis

April. 4
2017

Citation:
Goh, 2017

When They Talk about Momentum Method, What do They Talk About?

They talk about: why use the momentum; why it works better.

As “Why Momentum Really Works” implied, there is something else that is missing in most ML courses/posts.

- Partially due to the time/space limit in ML courses/posts

This is why taking an optimization course might be helpful.

I’ll mainly talk about how optimizers understand momentum method.

Outline

Review of GD Analysis

Starting from the simplest case:

2-dim diagonal quadratic

$$\min_{x_1,x_2\in\mathbb{R}} \frac{1}{2}(\lambda_1 x_1^2 + \lambda_2 x_2^2).$$

GD : $x^+ = x - \alpha \nabla f(x) = x - \alpha \begin{pmatrix} \lambda x_1 \\ \lambda x_2 \end{pmatrix}$.

$$\Rightarrow \begin{bmatrix} x_1^+ \\ x_2^+ \end{bmatrix} = \begin{bmatrix} (1-\alpha\lambda_1)x_1 \\ (1-\alpha\lambda_2)x_1 \end{bmatrix}$$

$x_1^+ = \gamma x_1$

$\Rightarrow x_1^k = \gamma x_1^{k-1} = \gamma^2 x_1^{k-2}$

$= \cdots = \gamma^k x_1^0$.

First eigen-mode rate: $|1-\alpha\lambda_1|$.

Second eigen-mode rate: $|1-\alpha\lambda_2|$.

Pick $\alpha = \frac{1}{\lambda_1}$, then the speed $\max\{0, |1-\frac{\lambda_2}{\lambda_1}|\} = 1 - \frac{1}{\kappa}$.

$= 1/L$

Analysis: Starting From 2-dim

Starting from the simplest case:

$$\min_{x_1,x_2\in\mathbb{R}} \frac{1}{2}(\lambda_1 x_1^2 + \lambda_2 x_2^2).$$

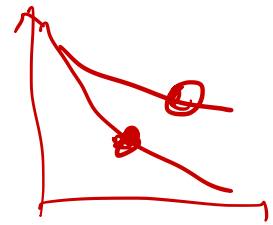
Heavy ball method:

$$x^+ = x - \alpha \nabla f(x) + \beta(x - x^-),$$

$$= \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} - \alpha \begin{pmatrix} \lambda_1 x_1 \\ \lambda_2 x_2 \end{pmatrix} + \beta \begin{pmatrix} x_1 - x_1^- \\ x_2 - x_2^- \end{pmatrix}.$$

$$\begin{bmatrix} x_1^+ \\ x_2^+ \end{bmatrix} = \begin{pmatrix} (1 - \alpha\lambda_1 + \beta) x_1 - \beta x_1^- \\ (1 - \alpha\lambda_2 + \beta) x_2 - \beta x_2^- \end{pmatrix}.$$

Analysis of 2-dim (cont'd)



Observation: The two sequences x_1^k and x_2^k are independent.

thus the convergence speed of x^k depends on worse speed.

Each sequence is like $\begin{pmatrix} x_1^k \\ x_2^k \end{pmatrix}$

$$a_{k+2} = \gamma a_{k+1} - \beta a_k, \tag{1}$$

where $\gamma = 1 - \alpha\lambda_i + \beta$.

e.g. Fibonacci sequence $a_{k+2} = a_{k+1} + a_k$,
$1, 1, 2, 3, 5, 8, 13, 21, 34, \ldots$.

$a_k = \frac{\alpha^k - \beta^k}{\alpha - \beta}$. $\alpha, \beta = \frac{1 \pm \sqrt{5}}{2} = \{1.618, 0.618\}$.

General formula:

$$a_k = c_1\sigma_1^k + c_2\sigma_2^k,$$

where σ_1, σ_2 are the two roots of

$$z^2 - \gamma z + \beta = 0.$$

Compare: $a_{k+2} - \gamma a_{k+1} + \beta a_k = 0$.

Question: For what α, β, $\{a_k\}$ converges?

Convergence of 2-term Recursion

$$a_k = c_1\sigma_1^k + c_2\sigma_2^k,$$

where $\sigma_{1,2} = \frac{\gamma \pm \sqrt{\gamma^2 - 4\beta}}{2}$ and $\gamma = 1 - \alpha\lambda_i + \beta$.

Case 1: $\gamma^2 - 4\beta > 0$. Two Real roots. Can ignore it. (too complicated)

Case 2: $\gamma^2 - 4\beta \leq 0$. Two conjugate complex roots.

Practice: in Case 2, for what α, β the sequence converges?

$|\sigma_1| < 1$, $|\sigma_2| < 1$.

Assume $\lambda_i = 1$.

Q1. $\gamma^2 - 4\beta \leq 0 \iff \alpha, \beta$?

Q2. If $\gamma^2 - 4\beta < 0$ holds, $|\sigma_1| < 1, |\sigma_2| < 1 \iff \alpha, \beta$?

Conclusion: When $\left|1 - \sqrt{\alpha\lambda_i}\right| \leq \sqrt{\beta} < 1$,

the sequence converges at rate $\sqrt{\beta}$.

The optimal speed is achieved when $\beta = (1 - \sqrt{\alpha\lambda_i})^2$, for a given i.

Remark: In the following derivations, assume $\alpha \geq 0$, $\beta \geq 0$.

Question: In Case 2 (i.e. $\gamma^2 \leq 4\beta$), for what $\alpha, \beta \geq 0$ the sequence converges?
Here $\gamma = 1 - \alpha + \beta$, the sequence is $a_{n+2} = \gamma a_{n+1} - \beta a_n$.

Answer: Decompose into two questions as below.

Q1. $\gamma^2 \leq 4\beta$, $\Leftrightarrow \alpha, \beta$?

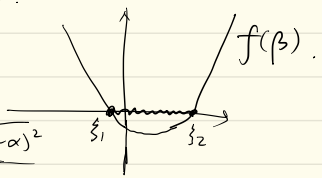
Recall $\gamma = 1 - \alpha + \beta$

Then $(1-\alpha+\beta)^2 \leq 4\beta$

$\Leftrightarrow \beta^2 + 2\beta(1-\alpha) + (1-\alpha)^2 - 4\beta \leq 0$.

$\Leftrightarrow \beta^2 - 2\beta(1+\alpha) + (1-\alpha)^2 \leq 0$.

quadratic function of β.



two roots are

$$\beta_{1,2} = \frac{2(1+\alpha) \pm \sqrt{4(1+\alpha)^2 - 4(1-\alpha)^2}}{2}$$

$$= 1+\alpha \pm 2\sqrt{\alpha} = (1 \pm \sqrt{\alpha})^2$$

$\therefore$ $\boxed{|1-\sqrt{\alpha}| \leq \sqrt{\beta} \leq 1+\sqrt{\alpha}}$ ①

Q2. If $\gamma^2-4\beta\le 0$ holds,
then $|\sigma_1|, |\sigma_2|<1 \iff \alpha, \beta$?

Recall the definition of the two roots

$$\sigma_{1,2} = \frac{\gamma \pm \sqrt{\gamma^2-4\beta}}{2}.$$

Since $\gamma^2-4\beta\le 0$, we know $\sqrt{\gamma^2-4\beta}$ is a complex numbe $\sqrt{4\beta-\gamma^2}\,i$, $i=\sqrt{-1}$.

Rewrite the two roots $\sigma_{1,2} = \frac{\gamma \pm \sqrt{4\beta-\gamma^2}\,i}{2}$

Their modulus

$$|\sigma_1| = |\sigma_2| = \sqrt{\left(\frac{\gamma}{2}\right)^2 + \left(\frac{\sqrt{4\beta-\gamma^2}}{2}\right)^2} = \frac{\sqrt{\gamma^2+(4\beta-\gamma^2)}}{2} = \sqrt{\beta}$$

$\therefore |\sigma_1|, |\sigma_2|<1 \iff \sqrt{\beta}<1.$

Magic! Only depend on β.

Conclusion. In Case 2, to guarantee convergence, need to have $|1-\sqrt{\alpha}| \le \beta < 1$.

2-dim Case

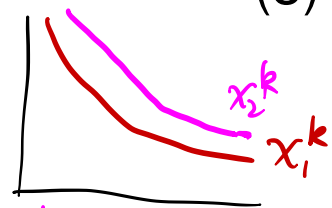
Now let's come back to the 2-dim case.

Two sequences x_1^k, x_2^k:

$$\{x_1^k\} \text{ converges if } |1 - \sqrt{\alpha\lambda_1}| \leq \sqrt{\beta} < 1, \tag{2}$$

$$\{x_2^k\} \text{ converges if } |1 - \sqrt{\alpha\lambda_2}| \leq \sqrt{\beta} < 1, \tag{3}$$

both with rate $\sqrt{\beta}$.



the same rate $\sqrt{\beta}$.

The overall convergence rate is optimal when

$$\sqrt{\beta} = \max\{|1 - \sqrt{\alpha\lambda_1}|, \ |1 - \sqrt{\alpha\lambda_2}|\}.$$

$\max\{0, 1 - \sqrt{\alpha\lambda_2}\} = 1 - \sqrt{\frac{\lambda_2}{\lambda_1}}$

Pick $\alpha = \frac{1}{\lambda_1} = 1/L$, then the optimal $\beta = \left(1 - \sqrt{\frac{\lambda_2}{\lambda_1}}\right)^2$, and the rate is

$$\sqrt{\beta} = 1 - \sqrt{\frac{\lambda_2}{\lambda_1}} = 1 - \sqrt{\frac{1}{\kappa}}.$$

Notice two things: (1) κ appears again; (2) additional "$\sqrt{\ }$"

Result for Quadratic Problem

The analysis for 2-dim can be extended to general n-dim strongly convex quadratic problem.

Proposition 3.1: Suppose Q is symmetric PD. Consider solving

$$\min_{\mathbf{x}\in\mathbb{R}^n} f(\mathbf{x}) \triangleq \mathbf{x}^T Q\mathbf{x} + 2\mathbf{b}^T x + \mathbf{c}.$$

Use heavy ball method with

$$\alpha = \frac{1}{L}, \quad \beta = (1-\sqrt{1/\kappa})^2,$$

where $L = \lambda_{\max}(Q)$, $\kappa = \frac{\lambda_{max}(Q)}{\lambda_{min}(Q)}$ is the condition number.

Then the asymptotic convergence rate $\frac{\|x^r - x^*\|}{\|x^0 - x^*\|} \le (1 - \frac{1}{\sqrt{\kappa}})^r \cdot C$, for some constant C.

$$\exp(\lim_{r\to\infty} \frac{1}{r} \log(\|x^r - x^*\|/\|x^0 - x^*\|)) = 1 - \sqrt{1/\kappa}.$$

Remark: To achieve relative error $\frac{\|\mathbf{x}^r - x^*\|}{\|\mathbf{x}^0 - x^*\|} < \epsilon$, only need # of iterations

$$\sqrt{\kappa}\log(\frac{1}{\epsilon}).$$

Theoretical Implication

$$x^+ = x - \frac{1}{L}\nabla f(x) + (1 - \sqrt{1/\kappa})^2(x - x^-).$$

Original GD: $\tilde{O}(\kappa)$ iterations. (ignore $\log 1/\epsilon$ term).

Adding a momentum term reduces to $\tilde{O}(\sqrt{\kappa})$ iterations.

Accelerate by $\sqrt{\kappa}$ times.

- Example: $\kappa = 10^4$, accelerate by 100 times.

In practice, hard to observe 100-time speedup, but somewhere between $[1, \sqrt{\kappa}]$.

This is the major reason why people believe momentum can help!

Some Drawbacks

Drawback 1: For non-quadratic function, such acceleration is lost – at least no one knows how to pick α, β to achieve it.

Drawback 2: Nonconvex problems? Unknown.

Engineers:
Drawback 3: Two parameters α, β to tune in practice. Harder to tune.

Drawback 4: oscillating.

due to complex eigenvalues.

End of Tuesday Feb 6 Class.

Better Stepsize for GD

Come back to 2-dim case.

$$\begin{bmatrix} x_1^+ \\ x_2^+ \end{bmatrix} = \begin{bmatrix} (1-\alpha\lambda_1)x_1 \\ (1-\alpha\lambda_2)x_1 \end{bmatrix} \qquad \begin{matrix} |1-\alpha\lambda_1| \\ |1-\alpha\lambda_2| \end{matrix}$$

The rate $\max\{|1-\alpha\lambda_1|, |1-\alpha\lambda_2|\}$. Pick $\alpha = \frac{1}{\lambda_1}$, the rate $1 - \frac{1}{\kappa}$.

Can we do better?

Yes. Let $|1-\alpha\lambda_1| = |1-\alpha\lambda_2|$, i.e.

$$\alpha = \frac{2}{\lambda_{\max} + \lambda_{\min}},$$

then the rate is

$$\frac{\lambda - 1}{\lambda + 1} = \frac{\kappa - 1}{\kappa + 1} = 1 - \frac{2}{\kappa + 1}.$$

# of iterations about $\frac{\kappa+1}{2}\log(1/\epsilon)$, twice faster than stepsize $1/L$.

Better Stepsize for GD

Come back to 2-dim case.

$1-\frac{1}{2}$: $2\log\frac{1}{\epsilon}$ iterations

$1-\frac{1}{10}$: $10\log\frac{1}{\epsilon}$ - - -

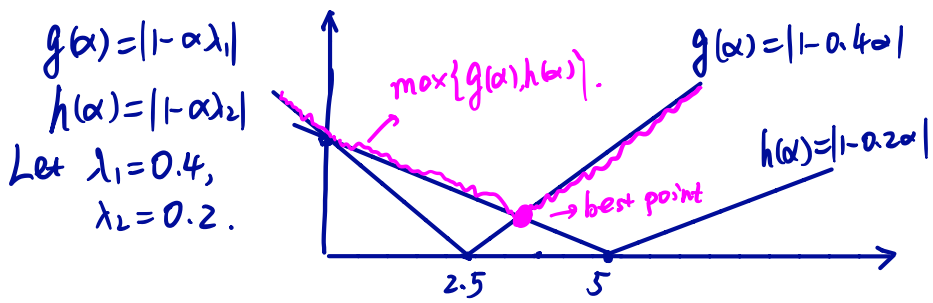
$1-\frac{2}{10}$: $\frac{10}{2}\log\frac{1}{\epsilon}$ - - .

$$\begin{bmatrix} x_1^+ \\ x_2^+ \end{bmatrix} = \begin{bmatrix} (1-\alpha\lambda_1)x_1 \\ (1-\alpha\lambda_2)x_1 \end{bmatrix}$$

The rate $\max\{|1-\alpha\lambda_1|, |1-\alpha\lambda_2|\}$. Pick $\alpha = \frac{1}{\lambda_1}$, the rate $1-\frac{1}{\kappa}$.

Can we do better?

Yes. Let $|1-\alpha\lambda_1| = |1-\alpha\lambda_2|$, i.e.



$$\alpha = \frac{2}{\lambda_{\max}+\lambda_{\min}},$$

GD with stepsize $\frac{1}{L}$: $1-\frac{1}{\kappa}$.

then the rate is

$$\frac{\lambda-1}{\lambda+1} = \frac{\kappa-1}{\kappa+1} = 1-\frac{2}{\kappa+1} \cdot \approx 1-\frac{2}{\kappa}$$

# of iterations about $\frac{\kappa+1}{2}\log(1/\epsilon)$, twice faster than stepsize $1/L$.

Better Stepsize for GD with Momentum

Two sequences x_1^k, x_2^k:

$$\{x_1^k\} \text{ converges if } |1 - \sqrt{\alpha\lambda_1}| \leq \sqrt{\beta} < 1, \tag{4}$$

$$\{x_2^k\} \text{ converges if } |1 - \sqrt{\alpha\lambda_2}| \leq \sqrt{\beta} < 1, \tag{5}$$

both with rate $\sqrt{\beta}$.

The overall convergence rate is optimal when $|1 - \sqrt{\alpha}\lambda_1| = |1 - \sqrt{\lambda_2\alpha}|$.

$$\sqrt{\beta} = \max\{|1 - \sqrt{\alpha\lambda_1}|, \ |1 - \sqrt{\alpha\lambda_2}|\}.$$

Pick $\alpha = \left(\frac{2}{\sqrt{\lambda_1} + \sqrt{\lambda_2}}\right)^2$, then the optimal $\beta = \left(\frac{\sqrt{\lambda_1} - \sqrt{\lambda_2}}{\sqrt{\lambda_1} + \sqrt{\lambda_2}}\right)^2 = \left(\frac{\sqrt{\kappa} - 1}{\sqrt{\kappa} + 1}\right)^2$, and the rate is

$$\sqrt{\beta} = 1 - \frac{2}{\sqrt{\kappa} + 1}.$$

HB with $\alpha = \frac{1}{L}$: rate $1 - \frac{1}{\sqrt{\kappa}}$.

Summary: Optimal Stepsize

- The analysis for 2-dim can be extended to more general case.
- Use GD to solve strongly convex problem, the optimal stepsize is

$$\alpha = \frac{2}{L+\mu}.$$

Convergence rate is $\frac{\kappa-1}{\kappa+1}$. $= 1 - \frac{2}{\kappa+1}$

# of iterations is at most $\frac{\kappa+1}{2}\log\frac{1}{\epsilon}$. $\approx \frac{\kappa}{2}$.

- Use GD with momentum (heavy ball method) to solve strongly convex quadratic problem, the optimal stepsize is

$$\alpha = \left(\frac{2}{\sqrt{\lambda_{\max}}+\sqrt{\lambda_{\min}}}\right)^2, \quad \beta = \left(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right)^2.$$

Convergence rate is $\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}$. $= 1 - \frac{2}{\sqrt{\kappa}+1}$

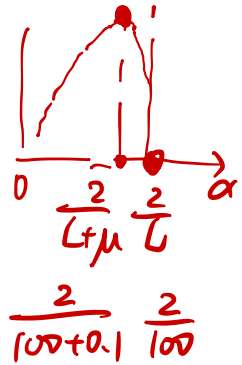
# of iterations is at most $\frac{\sqrt{\kappa}+1}{2}\log\frac{1}{\epsilon}$. $\approx \frac{\sqrt{\kappa}}{2}$.

Question: If need one day to tune α, how many days to tune two parameters α, β?

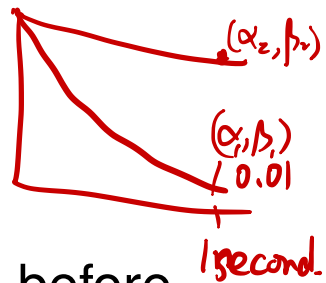
Answer: If try N parameters in 1 day for α, then for α, β, both trying N parameters, total N^2 combinations. So N days.

Practical Stepsize Choice

These are theoretical stepsizes; any practical guidance to engineers?



- GD: Theoretical stepsize $2/(L+\mu)$; diverge for $> 2/L$, so...
- **GD practice**: "knife's edge". Slightly smaller than the converge/diverge threshold

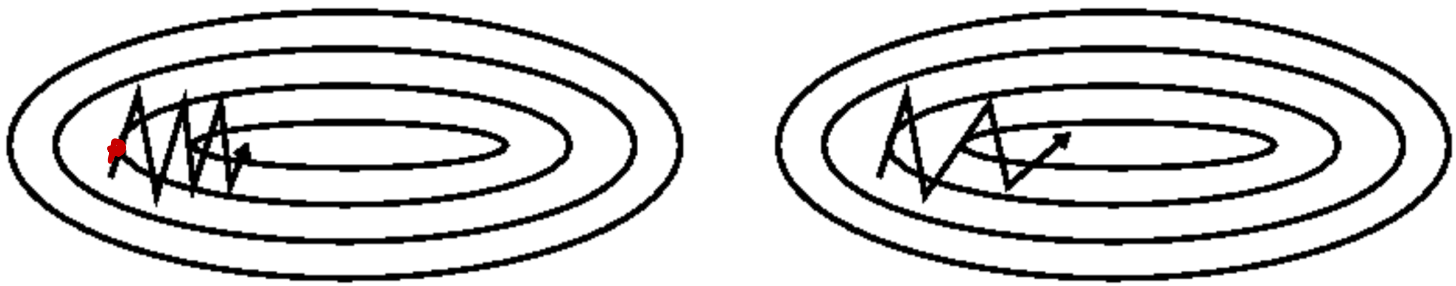


- Heavy ball (HB): theoretical α slightly smaller than $4/L$, $\beta \approx (1 - \frac{2}{\sqrt{\kappa}})^2$
- **HB Practice**: pick β slightly smaller than 1, and tune α up before divergence ("knife's edge") e.g. $\beta = 0.9, 0.99$.

Remark: These guidelines work for convex case; for nonconvex, may or may not work, but can be a starting point to try.

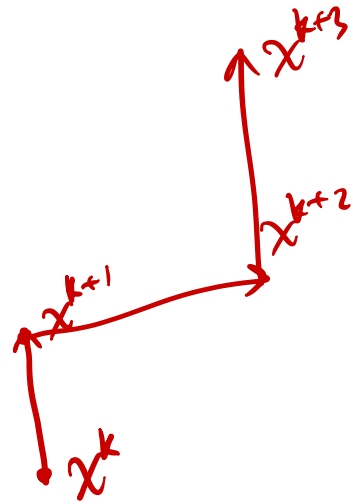
System Designer's Motivation: History Helps

Explanation why momentum works: reduce "zigzag" behavior.



Nature's Motivation: Momentum in Driving

Recall: GD is doing hill clibming (in a greedy way).



What if you are driving uphill?

Newton's low: $F=ma$, acceleration only changes velocity

Difference: You change velocity, insteading of changing position directly.

Polyak: heavy ball moving according to neg-gradient; but due to"inertia", it has momentum.

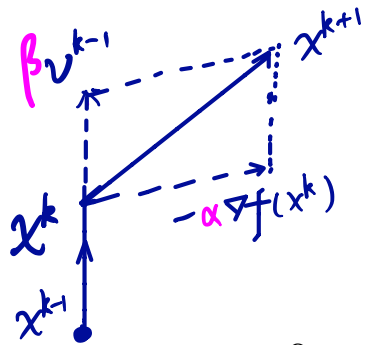
Changing Speed by Gradient

v: velocity, x: position.

$$x^{k+1} = x^k + 1 \cdot v^k, \qquad \text{----(1a)}$$

$$v^k = \beta v^{k-1} - \alpha \nabla f(x^k). \qquad \text{---(1b)}$$

Graph:



This is equivalent to (set $v^0 = 0$)

Heavy ball method : $x^{k+1} = x^k - \alpha \nabla f(x^k) + \beta(x^k - x^{k-1})$. ------(2)

Task: Find the three coefficients in (1), such that (1) $\Longleftrightarrow$ (2).

Hint 1: Let the three coefficients be 1, then try to simplify (1) to one equation.

Hint 2: Eliminate v's in (1), since (2) has no "v".

Major difference: (1) has v;

(2) has no v.

From (1) to (2), the task is: eliminate v.

$$\begin{cases} x^{k+1} = x^k + v^k & -- (1a) \\ v^k = v^{k-1} - \nabla f(x^k) & (1b) \end{cases}$$

Eliminate v from (1a): $v^k = x^{k+1} - x^k$.

Plug into (1b): $(x^{k+1} - x^k) = (x^k - x^{k-1}) - \nabla f(x^k)$. (3)

Compare with Heavy Ball method:

need to add β and α to the equation (3), i.e.

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

Thus (1b) should be changed to

$$v^k = \beta v^{k-1} - \alpha \nabla f(x^k);$$

(1a) is kept the same.

Another way to quickly derive (1) from (2):

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

$$\Rightarrow \quad \underbrace{x^{k+1} - x^k}_{=v^k} = \beta \underbrace{(x^k - x^{k-1})}_{=v^{k-1}} - \alpha \nabla f(x^k).$$

$$\Rightarrow \quad v^k = x^{k+1} - x^k, \quad \text{and} \quad v^{k+1} = v^k - \alpha \nabla f(x^k). \Rightarrow (1).$$

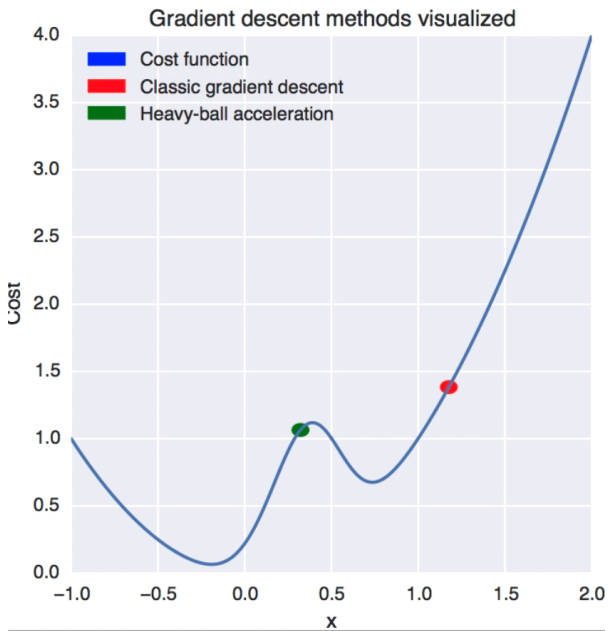
Does Momentum Work Better for Nonconvex Problems?

A popular claim: momentum helps avoid local-min.

Polyak'74 argued: for GD $x^{k+1} = x^k - \alpha \nabla f(x^k)$, when $\nabla f(x^k)$ is small, the ball will stay in the "valley".

But for HB, the extra $x^k - x^{k-1}$ term may pull the ball out of the basin.

Can be a research topic or course project.



Appendix: Convergence of 2-term Recursion (skip in class)

$$a_k = c_1\sigma_1^k + c_2\sigma_2^k,$$

where $\sigma_{1,2} = \frac{\gamma \pm \sqrt{\gamma^2 - 4\beta}}{2}$ and $\gamma = 1 - \alpha\lambda_i + \beta$.

First question: when does the sequence converge?
Answer:

$$\max\{|\sigma_1|, |\sigma_2|\} < 1.$$

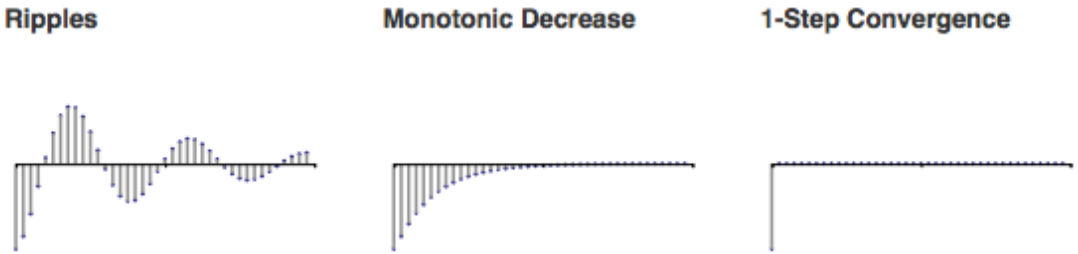
A sufficient condition for convergence is

$$0 \leq \beta < 1, \quad 0 < \alpha\lambda_i < 2 + 2\beta.$$

Remark: When β is close to 1, stepsize $\alpha < 4/\lambda_i$; 2 times larger than GD upper bound $2/\lambda_i$.

Appendix: Fastest Convergence Rate (skip in class)

Second question: when does the sequence converge the fastest?



- If α and β are both free, the best choice is

$$\beta = 0, \quad \alpha = 1/\lambda_i.$$

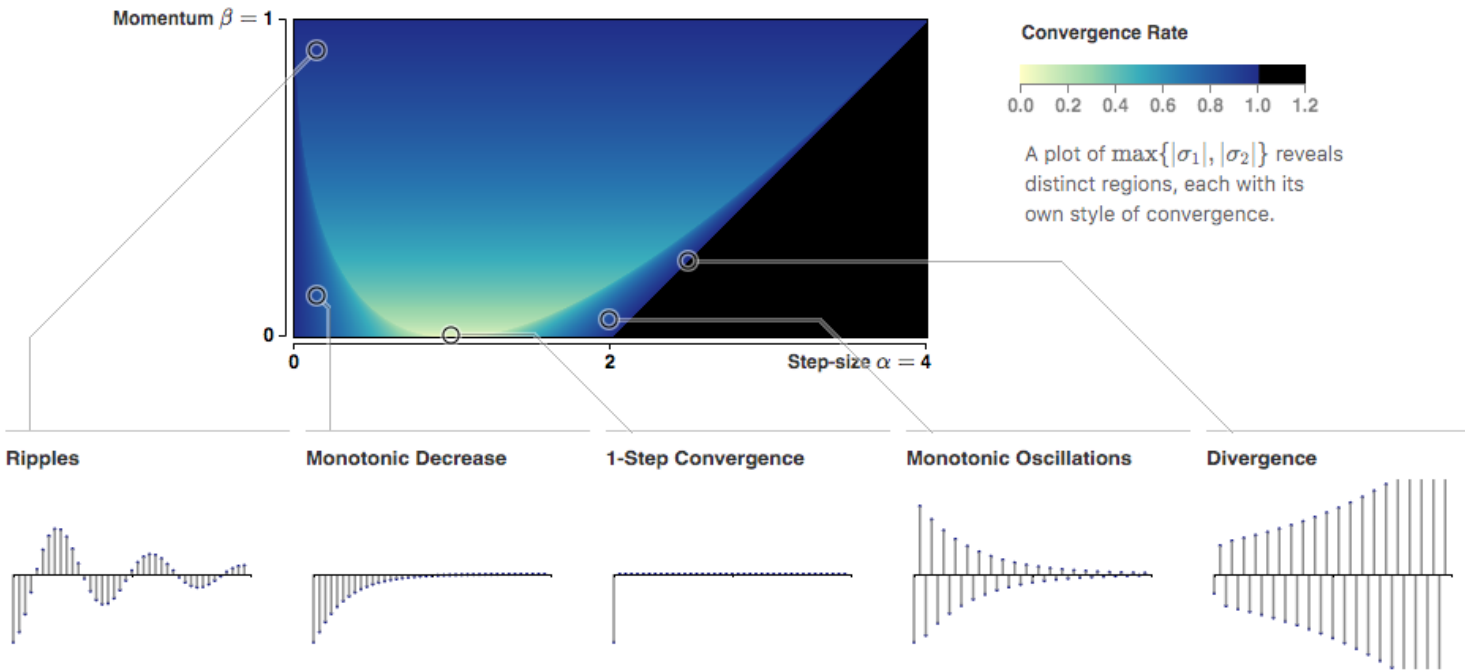
  1 step converge.

- But α is not completely free (why?)

- Best $\beta = (1 - \sqrt{\alpha\lambda_i})^2$, and converge at rate $\sqrt{\beta}$.
  - This happens when $\sigma_1 = \sigma_2$, i.e., $\gamma^2 - 4\beta = (1 - \alpha\lambda_i + \beta)^2 - 4\beta = 0$.

Appendix: Convergence Rate v.s. Parameter Choice (skip in class)

Figures of convergence rate v.s. α, β. Here α replaces $\alpha\lambda_i$.



Two boundaries: $\alpha = 2\beta + 2$; $\beta = (1 - \sqrt{\alpha})^2$.

Conclusion of Today

Can you summarize yourself?

- Heavy ball method adds a momentum term to GD
- It works better because faster by a factor of $\sqrt{\kappa}$, for (strongly convex) quadratic problem
- Practical choice of stepsize: β slightly smaller than 1, and α as large as possible

Conclusion of Today

Can you summarize yourself?

- Heavy ball method adds a momentum term to GD
- It works better because faster by a factor of $\sqrt{\kappa}$, for (strongly convex) quadratic problem
- Practical choice of stepsize: β slightly smaller than 1, and α as large as possible