First-Order Methods

Yueze Fu 2022-12-24

First-Order Methods

- In this section, we consider iterative optimization methods that leverage first-order derivatives of the objective function. These methods perform an update of the following form:

  $$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t + \eta_t \boldsymbol{d_t}$$

- d_t is a **descent direction**, and η_t is known as the **step size** or **learning rate**.

PART 1

8.2.1 Descent direction

8.2.1 Descent direction

- We say that a direction $\boldsymbol{d}$ is a descent direction if there is a small enough (but nonzero) amount $\boldsymbol{\eta}$ we can move in direction $\boldsymbol{d}$ and be guaranteed to decrease the function value .

$$\mathcal{L}(\boldsymbol{\theta}+\eta\boldsymbol{d})<\mathcal{L}(\boldsymbol{\theta}) \quad \eta>0$$

$$\boldsymbol{g}_t \triangleq \nabla\mathcal{L}(\boldsymbol{\theta})|_{\boldsymbol{\theta}_t}=\nabla\mathcal{L}(\boldsymbol{\theta})=\boldsymbol{g}(\boldsymbol{\theta}_t)$$

- This points in the direction of maximal increase in f, so the negative gradient is a descent direction. It can be shown that any direction $\boldsymbol{d}$ is also a descent direction if the angle θ between $\boldsymbol{d}$ and $-\boldsymbol{g}_t$ is less than 90 degrees and satisfies

$$\boldsymbol{d}^T\boldsymbol{g}_t=||\boldsymbol{d}||\,||\boldsymbol{g}_t||\cos(\theta)<0$$

8.2.1 Descent direction

- The form of the iteration function is as follows:

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t + \eta_t \boldsymbol{d_t}$$

- The loss function can be wrote as:

$$\mathcal{L}(\boldsymbol{\theta}_{t+1}) = \mathcal{L}(\boldsymbol{\theta}_t + \eta \boldsymbol{d}_t)$$

- Use Taylor Formula at point

$$\mathcal{L}(\boldsymbol{\theta}_{t+1}) = \mathcal{L}(\boldsymbol{\theta}_t) + \nabla f(\theta_t)^{\boldsymbol{T}} \eta \boldsymbol{d}_t + O(\eta \boldsymbol{d}_t)$$

$$\mathcal{L}(\boldsymbol{\theta}_{t+1}) - \mathcal{L}(\boldsymbol{\theta}_t) = \nabla f(\theta_t)^{\boldsymbol{T}} \eta \boldsymbol{d}_t + O(\eta \boldsymbol{d}_t) < 0$$

- η > 0 and ignore the remainder:

$$\nabla f(\theta_t)^{\boldsymbol{T}} \boldsymbol{d}_t < 0$$

$$\boldsymbol{d}^{\boldsymbol{T}} \boldsymbol{g}_t = ||\boldsymbol{d}||\,||\boldsymbol{g}_t||\cos(\theta) < 0$$

$\boldsymbol{d_t} = -\boldsymbol{g_t}$
Steepest descent

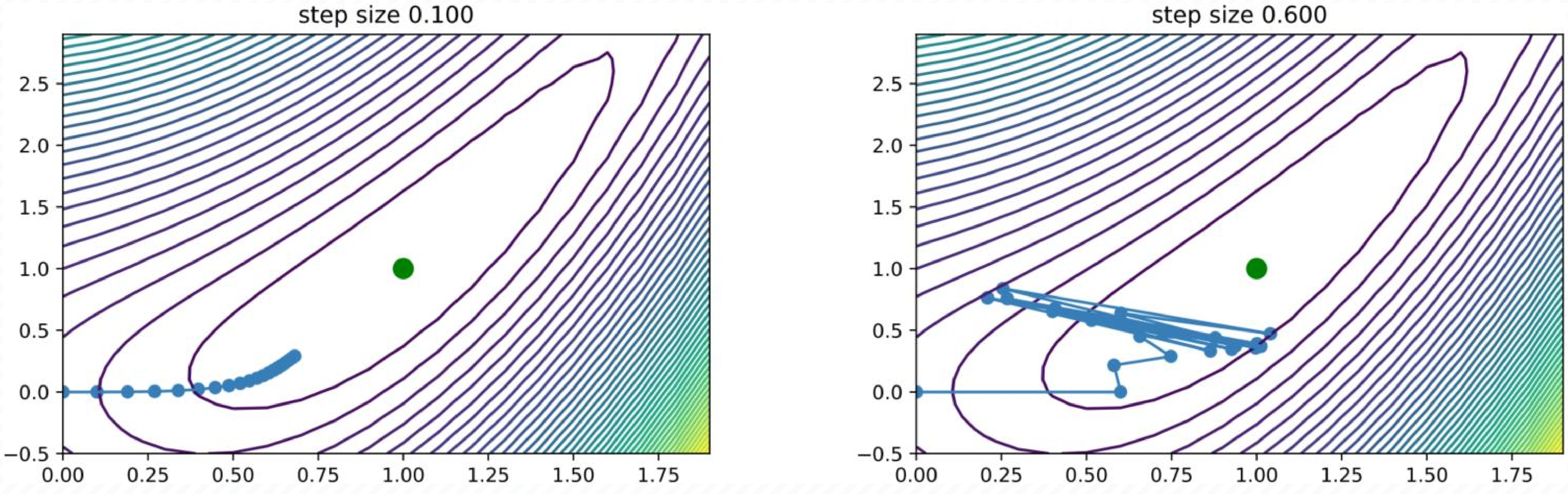
PART 2

8.2.2 Step size

8.2.2.1 Constant step size

✍ The simplest method is to use a constant step size, $\eta_t = \eta$. However, if it is too large, the method may fail to converge, and if it is too small, the method will converge but very slowly. For example:

$$\mathcal{L}(\boldsymbol{\theta}) = 0.5(\theta_1^2 + \theta_2)^2 + 0.5(\theta_1 - 1)^2$$



8.2.2.1 Constant step size

In some cases, we can derive a theoretical upper bound on the maximum step size we can use. For example, consider a quadratic objective

$$\mathcal{L}(\boldsymbol{\theta}) = \frac{1}{2}\boldsymbol{\theta}^{\boldsymbol{T}}\boldsymbol{A}\boldsymbol{\theta} + \boldsymbol{b}^{\boldsymbol{T}}\boldsymbol{\theta} + c$$

One can show that steepest descent will have global convergence if the step size satisfies:

$$\eta < \frac{2}{\lambda_{\max}(\boldsymbol{A})}$$

More generally, we can set:

$$\eta < \frac{2}{L}$$

8.2.2.2 Line Search

✍ The optimal step size can be found by finding the value that maximally decreases the objective along the chosen direction by solving the 1d minimization problem

$$\eta_t = arg\min_{\eta>0} \phi_t(\eta) = arg\min_{\eta>0} \mathcal{L}(\boldsymbol{\theta_t} + \eta \boldsymbol{d_t})$$

✍ This is known as **line search**, since we are searching along the line defined by $\boldsymbol{d_t}$.

$\boldsymbol{\theta_t}$ and $\boldsymbol{d_t}$ are fixed. For example, consider the quadratic loss

$$\mathcal{L}(\boldsymbol{\theta}) = \frac{1}{2}\boldsymbol{\theta^T A \theta} + \boldsymbol{b^T \theta} + c$$

✍ Computing the derivative of ϕ gives

$$\frac{d\phi(\eta)}{d\eta} = \frac{d}{d\eta}\left[\frac{1}{2}(\boldsymbol{\theta} + \eta\boldsymbol{d})^T A(\boldsymbol{\theta} + \eta\boldsymbol{d}) + \boldsymbol{b^T}(\boldsymbol{\theta} + \eta\boldsymbol{d}) + c\right] \qquad \frac{d\phi(\eta)}{d\eta} = 0 \Leftrightarrow \eta = -\frac{\boldsymbol{d^T}(\boldsymbol{A\theta} + \boldsymbol{b})}{\boldsymbol{d^T A d}}$$

$$= \boldsymbol{d^T A}(\boldsymbol{\theta} + \eta\boldsymbol{d}) + \boldsymbol{d^T b}$$

$$= \boldsymbol{d^T}(\boldsymbol{A\theta} + \boldsymbol{b}) + \eta\boldsymbol{d^T A d}$$

8.2.2.2 Line Search

- Using the optimal step size is known as **exact line search**. However, it is not usually necessary to be so precise. We can start with the current stepsize (or some maximum value), and then reduce it by a factor $0 < \beta < 1$ at each step until we satisfy the following condition, known as the **Armijo-Goldstein** test:

$$\mathcal{L}(\boldsymbol{\theta}_t + \eta \boldsymbol{d}_t) \leqslant \mathcal{L}(\boldsymbol{\theta}_t) + c\eta \boldsymbol{d}_t^T \nabla \mathcal{L}(\boldsymbol{\theta}_t)$$

Typically c = 10^{-4}

PART 3

8.2.3 Convergence rates

8.2.3 Convergence rates

- For certain convex problems, with a gradient with bounded Lipschitz constant, one can show that gradient descent converges at a linear rate. This means that there exists a number $0 < \mu < 1$ such that

  $$|\mathcal{L}(\boldsymbol{\theta}_{t+1}) - \mathcal{L}(\boldsymbol{\theta}_*)| \leq \mu |\mathcal{L}(\boldsymbol{\theta}_t) - \mathcal{L}(\theta_*)|$$

  Here μ is called the **rate of convergence**.

- For some simple problems, we can derive the convergence rate explicitly

  $$\mathcal{L}(\boldsymbol{\theta}) = \frac{1}{2}\boldsymbol{\theta}^T \boldsymbol{A}\boldsymbol{\theta} + \boldsymbol{b}^T \boldsymbol{\theta} + c$$

  $$\mu = \left(\frac{\lambda_{\max} - \lambda_{\min}}{\lambda_{\max} + \lambda_{\min}}\right)^2 \xrightarrow{\text{rewrite}} \mu = \left(\frac{k-1}{k+1}\right)^2 \quad k = \frac{\lambda_{\max}}{\lambda_{\min}}$$

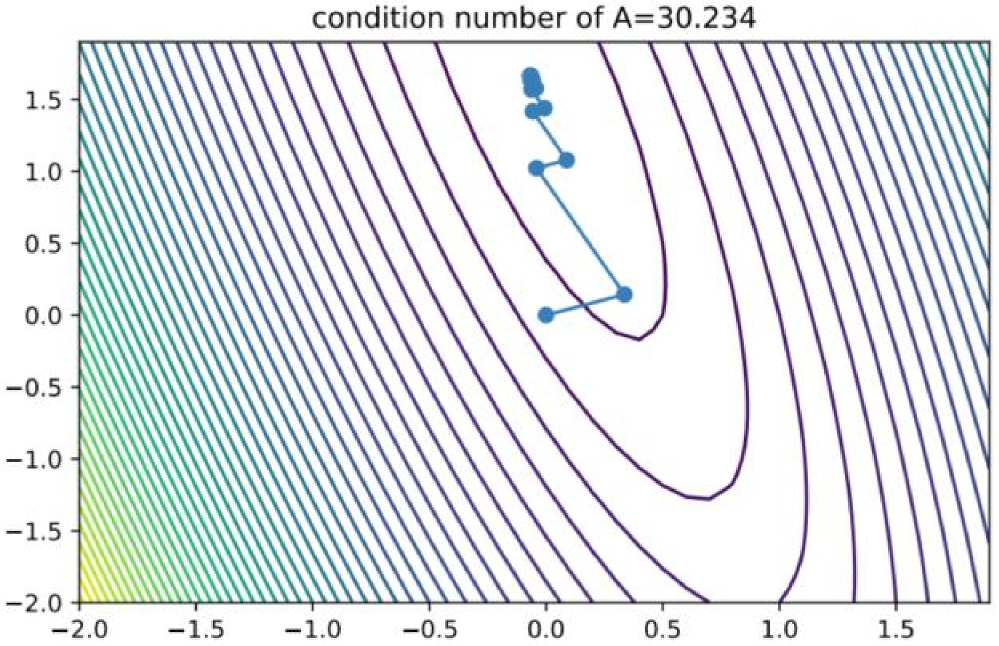
  Condition number

R: 证明以及相关信息，感谢周绪睿同学的分享

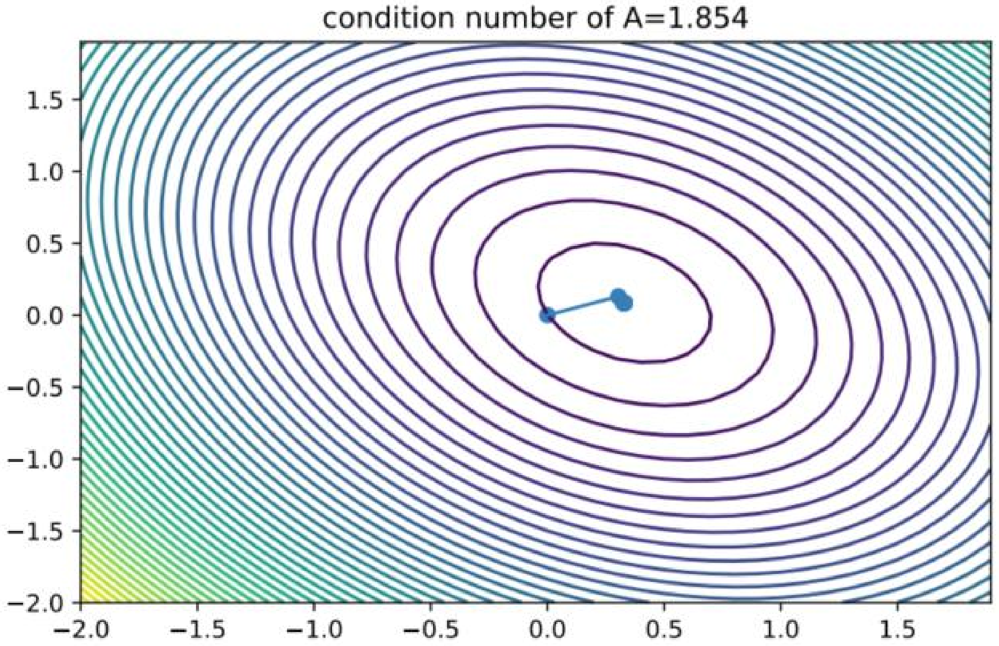
8.2.3 Convergence rates

✍ An example

A = [20,5;5,2], b = [-14;-6] c = 10



A = [20,5;5,16], b = [-14;-6] c = 10

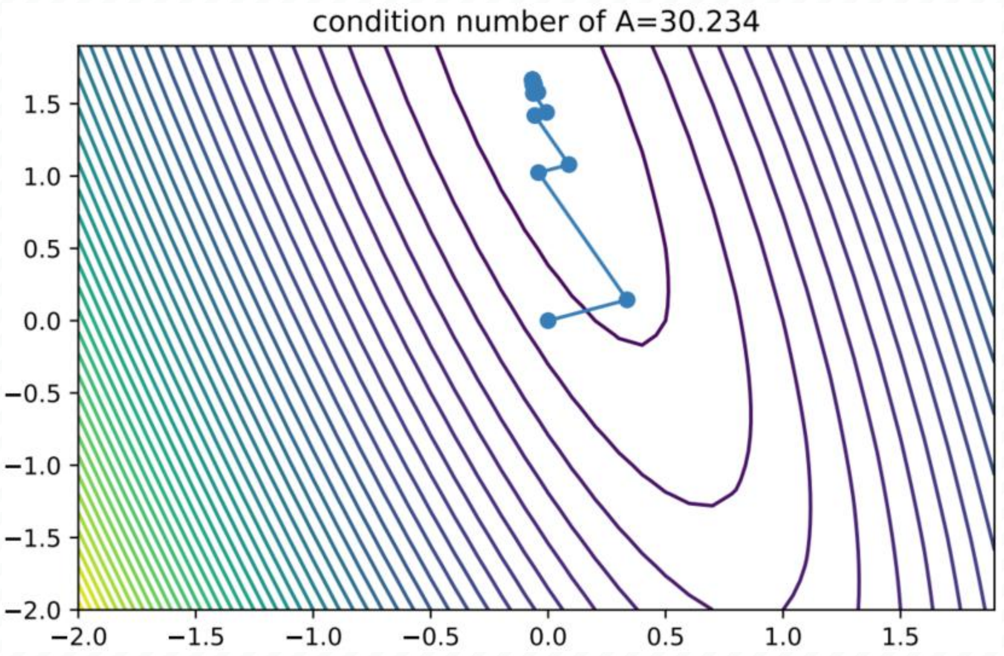


We see that steepest descent converges much more quickly for the problem with the smaller condition number.

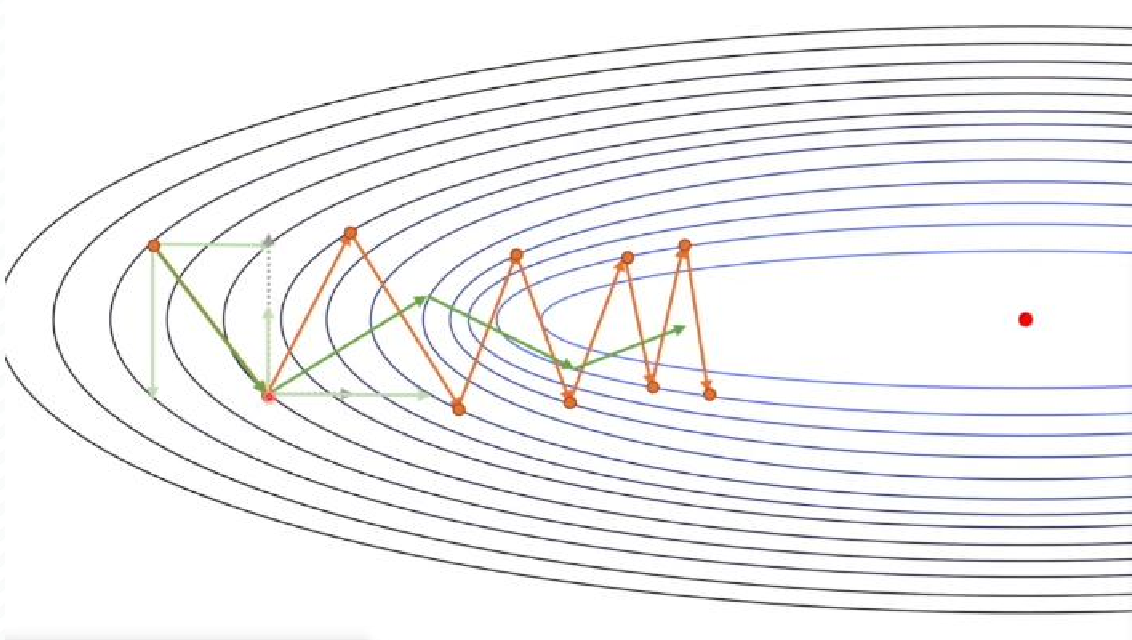
PART 4

8.2.4 Momentum methods

8.2.4 Momentum methods



- Gradient descent can move very slowly along flat regions of the loss landscape.
- We can analyze the figure to find the reason.



- Intuitively, we can think that this oscillation is caused by components on the vertical axis.

8.2.4.1 Momentum

- ✍ One simple heuristic, known as the momentum method, is to move faster along directions that were previously good, and to slow down along directions where the gradient has suddenly changed. This can be implemented as follows:

$$\boldsymbol{m_t} = \beta \boldsymbol{m_{t-1}} + \boldsymbol{g_{t-1}}$$

$$\boldsymbol{\theta_t} = \boldsymbol{\theta_{t-1}} - \eta_t \boldsymbol{m_t}$$

- ✍ where m_t is the momentum and $0 < \beta < 1$. A typical value of β is 0.9.

$$\boldsymbol{m_t} = \beta \boldsymbol{m_{t-1}} + \boldsymbol{g_{t-1}} = \beta^2 \boldsymbol{m_{t-2}} + \beta \boldsymbol{g_{t-2}} + \boldsymbol{g_{t-1}} = \ldots = \sum_{\tau=0}^{t-1} \beta^{\tau} \boldsymbol{g_{t-\tau-1}}$$

8.2.4.2 Nesterov momentum

- One problem with the standard momentum method is that it may not slow down enough at the bottom of a valley, causing oscillation.

Nesterov accelerated gradient method:

$$\tilde{\boldsymbol{\theta}}_{t+1} = \boldsymbol{\theta}_t + \beta(\boldsymbol{\theta}_t - \boldsymbol{\theta}_{t-1})$$

$$\boldsymbol{\theta}_{t+1} = \tilde{\boldsymbol{\theta}}_{t+1} - \eta_t \nabla \mathcal{L}\left(\tilde{\boldsymbol{\theta}}_{t+1}\right)$$

rewrite

$$\boldsymbol{m}_{t+1} = \beta \boldsymbol{m}_t - \eta_t \nabla \mathcal{L}(\boldsymbol{\theta}_t + \beta \boldsymbol{m}_t)$$

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t + \boldsymbol{m}_{t+1}$$

contrast

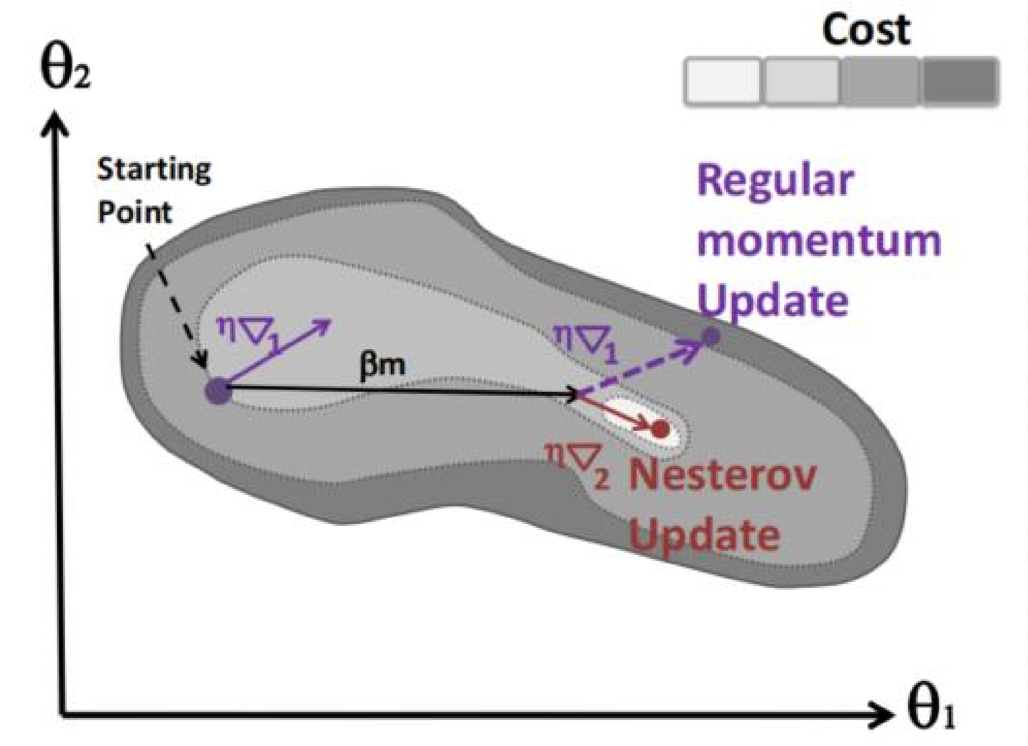
$$\boldsymbol{m}_t = \beta \boldsymbol{m}_{t-1} + \boldsymbol{g}_{t-1}$$

$$\boldsymbol{\theta}_t = \boldsymbol{\theta}_{t-1} - \eta_t \boldsymbol{m}_t$$

- This is essentially a form of one-step "look ahead", that can reduce the amount of oscillation.

8.2.4.2 Nesterov momentum

- The momentum vector is already roughly pointing in the right direction, so measuring the gradient at the new location $\boldsymbol{\theta_t}+\beta \boldsymbol{m_t}$, rather than the current location $\boldsymbol{\theta_t}$, can be more accurate.



- The Nesterov accelerated gradient method is provably faster than steepest descent for convex functions when β and η_t are chosen appropriately.

- In practice, however, using Nesterov momentum can be slower than steepest descent, and can even unstable if β or $\eta_t$are misspecified.

End

Thank you for your listening!