Session 12: Gradient descent

Optimization and Computational Linear Algebra for Data Science

Léo Miolane

Contents

Gradient descent

Gradient descent algorithm

Goal: minimize a differentiable function $f : \mathbb{R}^n \to \mathbb{R}$.

Starting from a point $x_0 \in \mathbb{R}^n$, perform the updates:

$$x_{t+1} = x_t - \alpha_t \nabla f(x_t).$$

Convex vs non-convex

Convex

Non-convex

Numerical observations

- If the step size α is small enough, gradient descent converges to $x^\star$ **but** this may take a while.
- If the step size α is large, gradient descent moves faster **but** it may oscilate or even diverge.
- The convergence is faster when the eigenvalues of the Hessian H_f are of close to each other.

Convergence analysis for convex functions

Smoothness and strong convexity

Definition

Given $L, \mu > 0$, we say that a twice-differentiable convex function $f : \mathbb{R}^n \to \mathbb{R}$ is

- L-smooth if for all $x \in \mathbb{R}^n$, $\lambda_{\max}(H_f(x)) \leq L$.
- μ-strongly convex if for all $x \in \mathbb{R}^n$, $\lambda_{\min}(H_f(x)) \geq \mu$.

Speed for L-smooth functions

Proposition

Assume that f is convex, L-smooth and admits a global minimizer $x^\star \in \mathbb{R}^n$. Then, gradient descent with constant step size $\alpha_t = 1/L$ verifies:

$$f(x_t) - f(x^\star) \leq \frac{2L\|x_0 - x^\star\|^2}{t+4}.$$

L-smooth + μ-strongly cvx functions

Theorem

Assume that f is convex, L-smooth and μ-strongly convex. Then, gradient descent with constant step size $\alpha_t = 1/L$ verifies:

$$f(x_t) - f(x^\star) \leq \left(1 - \frac{\mu}{L}\right)^t (f(x_0) - f(x^\star)).$$

Proof

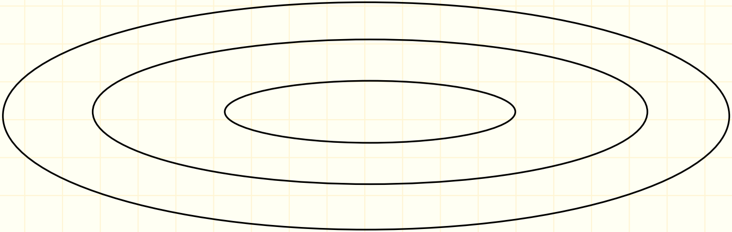
Choosing the step size

Backtracking line search

Start with $\alpha = 1$ and while

$$f(x_t - \alpha \nabla f(x_t)) \geq f(x_t) - \frac{\alpha}{2} \|\nabla f(x_t)\|^2,$$

update let's say $\alpha = 0.8\alpha$.

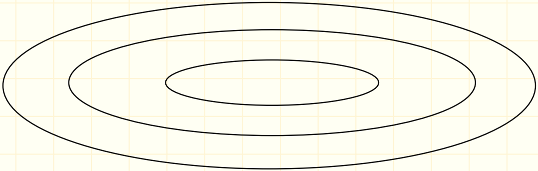


Improvements

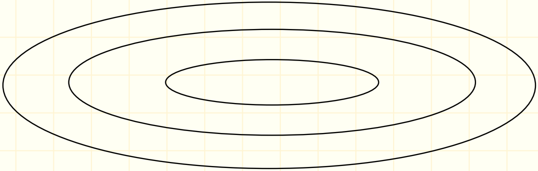
Issues with gradient descent

When the condition number $\kappa = L/\mu$ is large:

1. the norm $\|\nabla f(x)\|$ is sometimes too small.
   $\rightarrow$ gradient descent steps are too small.



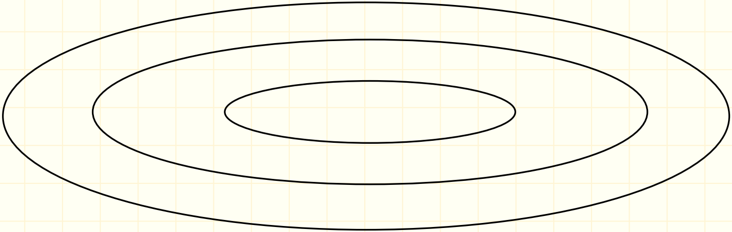
2. The vector $-\nabla f(x)$ does « not really » points towards the minimizer $x^\star$.
   $\rightarrow$ gradient descent oscilates.



Gradient descent + momentum

Idea: mimic the trajectory of an « heavy ball » that goes down the slope:

$$x_{t+1} = x_t + v_t \qquad \text{where} \quad v_t = -\alpha_t \nabla f(x_t) \; + \; \beta_t v_{t-1} \, .$$



Newton's method

Assume that f is μ-strongly convex and L-smooth.

Newton's method perform the updates:

$$x_{t+1} = x_t - H_f(x_t)^{-1} \nabla f(x_t).$$

Graphical interpretation

Advantages and drawbacks

- Extremly fast there exists $C, \rho > 0$ such that

$$\|x_t - x^\star\|^2 \leq Ce^{-\rho 2^t}.$$

- Computationally expensive: requires $\sim n^3$ operations to compute the inverse of the $n \times n$ matrix $H_f(x_t)$.
- In non-convex setting, Newton's method gets attracted by any critical points (which could be saddle points/maximas...).

Quasi-Newton methods: try to approximate $H_f(x_t)$ by matrices B_t that are easier to compute.

Questions?

Questions?