TP4 : CONVEX OPTIMIZATION

COURS D'APPRENTISSAGE, ECOLE NORMALE SUPÉRIEURE

Raphaël Berthier
raphael.berthier@inria.fr

1. Choice of the stepsizes for gradient descent in quadratic optimization

In this exercise, we are interested in minimizing a quadratic function, i.e., a function of the form

$$f(x) = \frac{1}{2}x^T Hx - b^T x\,,$$

where $H \succ 0$ is a $d \times d$ symmetric positive definite matrix and $b \in \mathbb{R}^d$. Denote x^* the minimum of f that we are aiming at. We use the gradient descent methods with stepsize γ :

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

1) Show the recurrence relation $x_{t+1} - x^* = (I_d - \gamma H)(x_t - x^*)$.

2) To study this recurrence relation, we diagonalize the matrix H : denote $\lambda_1 > \cdots > \lambda_d > 0$ its eigenvalues and $u_1, \dots, u_d$ the associated eigenvectors. What are the constants μ and L such that f is μ-strongly convexe and L-smooth ?

3) Show that

$$\langle x_t - x^*, u_i\rangle = (1 - \gamma\lambda_i)^t \langle x_0 - x^*, u_i\rangle\,.$$

4) What is the choice of γ maximizing the rate of convergence of $\|x_t - x^*\|_2$? Compare with the choice used in the lession.

In practice, the above rate of convergence can become very slow when the ratio μ/L is small. To solve this issue, we propose to use a method of the form

$$x_{t+1} = x_t - \gamma \nabla f(x_t) + \beta(x_t - x_{t-1})\,. \tag{1}$$

This iteration can be interpreted as follows : the speed $x_{t+1} - x_t$ at the time t is determined by the gradient—like for gradient descent—to which we add a fraction of the previous speed $x_t - x_{t-1}$ at the time $t-1$. This additional term accelerates show convergences and damps oscillatory behaviors.

The choice of the parameters γ and β can be optimized like for plain gradient descent. To simplify the computations, we directly give the optimal solution :

$$\gamma = \frac{4}{(\sqrt{L} + \sqrt{\mu})^2}\,, \qquad \beta = \left(\frac{\sqrt{L} - \sqrt{\mu}}{\sqrt{L} + \sqrt{\mu}}\right)^2\,. \tag{2}$$

5) Compute the second order recurrence relation satisfied by $\langle x_t - x^*, u_i\rangle$ and solve it. Give the rate of convergence of $\|x_t - x^*\|_2$.

The method defined by equations (1)-(2) is called the *heavy-ball* method because it can be interpreted as the discretization of the differential equation of a ball that would roll down the graph of the quadratic f. More generally, it is usual to choose x_{t+1} as a function of x_t, $\nabla f(x_t)$ *and of*

x_{t-1} in order to get accelerated optimization methods : one speaks about *inertial methods*. Although the heavy ball method does not necessarily work beyond quadratic functions, there are some simple modifications that do work on all (strongly convex) functions : see Nesterov's acceleration.

2. Gradient descent for ridge regression

We implement gradient descent with constant stepsize γ on a simple problem : ridge regression. We recall that it is the problem of minimizing the penalized quadratic empirical risk :

$$\min_{w \in \mathbb{R}^p} \frac{1}{2n} \|y - Xw\|_2^2 + \frac{\lambda}{2} \|w\|_2^2 \,.$$

We work in the regime $p > n$.

6) Show that this problem is of the form (1). What are the parameters H, b, L et μ ?

7) Recall the expression for the estimator of ridge regression (obtain it by calculus by canceling the gradient).

8) Generate randomly a design matrix $X \in \mathbb{R}^{n \times p}$ of size $n = 50, p = 60$ whose entries are i.i.d. standard Gaussian and a vector y also composed pf i.i.d. standard Gaussian random variables.

9) Fix an arbitrary value for λ, for instance $\lambda = 1$. Compute numerically μ and L. Represent though an histogram the eigenvalues of H.

10) We will now illustrate the convergence of a constant step size gradient descent method towards the optimum. Implement plain gradient descent to find the minimum value and the minimizer. Represent graphically the speed of convergence.

Remarks : - The speed of convergence are usually represented on logarithmic plots, please use the functions `semilogy`, `loglog` of `matplotlib.pyplot`.
- In practice, it might be hard to know the parameters μ and L. Thus one can not choose the theoretical recommendation for γ. In this case, γ become a "hand-tuned" hyperparameter.

11) Implement the heavy ball method and plot the speed of convergence.

12) What happens when the regularization parameter λ goes to 0 ?

This practical session show that in convex optimization, the hessian of the function to be minimized, and more specifically its condition number, is an essential parameter. In machine learning, these hessians are random because the data is random, and are of large dimension. The exists limit theorems that give properties that these matrices will satisfy with high probability ; the *random matrix theory* studies these properties. These properties can then be used by optimization algorithms for machine learning.