LAB1: The gradient descent method in action

We consider the problem of finding a minimizer of a convex smooth function $f\colon \mathbb{R}^d \to \mathbb{R}$; that is we want to solve

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

We assume that the function f has a Lipschitz continuous gradient. Note that the minimizer, when it exists, is not necessarily unique. The gradient descent method is defined as follows

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

where $\nabla f(x)$ is the gradient of f at x, $x^{(0)} \in \mathbb{R}^d$ is an arbitrary *initial point*. Convergence of the method is ensured if the *stepsize* γ satisfies

$$0 < \gamma < 2/L,$$

where L is the Lipschitz constant of ∇f, that is

$$(\forall\, x, y \in \mathbb{R}^d) \qquad \|\nabla f(x) - \nabla f(y)\| \leq L\,\|x - y\|\,.$$

If f is twice continuously differentiable, then

$$L = \sup_{x\in\mathbb{R}^d} \left\|\nabla^2 f(x)\right\|,$$

where $\nabla^2 f(x)$ is the Hessian of f at x (the matrix of partial second derivatives of f) and $\|\cdot\|$ is the (spectral) operator norm of f (largest eigenvalue).

1. Gradient Descent in 2D

We consider the function

$$f\colon \mathbb{R}^2 \to \mathbb{R}, \qquad f(x) = \frac{1}{2}(x_1^2 + \eta x_2^2), \tag{2}$$

where $\eta > 0$ controls the anisotropy of the problem. The function is twice continuously differentiable and its unique minimizer is zero. We assume $\eta > 1$. The Lipschitz constant of ∇f is $L = \eta$ and the function is strongly convex with modulus $\mu = 1$ (the smallest eigenvalue of $\nabla^2 f(x)$). So gradient descent *converges linearly* if

$$0 < \gamma < \frac{2}{\eta}.$$

(i) implement the gradient descent method (1) in Matlab as a function of $x^{(0)}$, η, and the maximum number of iterations.

(ii) Fix $\eta > 1$ (e.g. $\eta = 10$). Exploit different choices for γ (including $\gamma = 1/L$ and $\gamma = 2/(L+\mu)$) and recognize the following theoretical linear behavior

$$\left\|x^{(k)}\right\| \leq q(\gamma)^k \left\|x_0\right\|, \qquad q(\gamma) = \begin{cases} 1 - \gamma\mu & \text{if } \gamma \leq \frac{2}{L+\mu} \\ \gamma L - 1 & \text{if } \gamma \geq \frac{2}{L+\mu}. \end{cases}$$

Plot the values of $(\|x^{(k)}\|)_{k\in\mathbb{N}}$ and of $(q(\gamma)^k \left\|x^{(0)}\right\|)_{k\in\mathbb{N}}$.

(iii) Make a contour plot of the function f and the sequence $(x^{(k)})_{k\in\mathbb{N}}$.

(iv) Plot the values $(\|x^{(k)}\|)_{k\in\mathbb{N}}$ for several choices of γ in order to see how the convergence rate changes with γ and check that the choice $\gamma = 2/(L+\mu)$ is the best one.

(v) Fix $\gamma = 2/(L+\mu)$. Plot the trajectories of $(x^{(k)})_{k\in\mathbb{N}}$ for different values of η. Observe how the zig-zagging effect changes. Moreover, check the stopping rules

$$\|x_k - x_*\| \leq \frac{2}{\eta+1}\left(\frac{\eta-1}{\eta+1}\right)^k \|x_0 - x_1\| \leq \varepsilon \quad \text{and} \quad \|x_{k+1} - x_k\| \leq \varepsilon.$$

2. Linear Least Squares Problems

We consider the problem of minimizing

$$f(x) = \frac{1}{2}\|Ax - y\|^2, \qquad A \in \mathbb{R}^{n\times m} \text{ and } y \in \mathbb{R}^n. \tag{3}$$

The case $n > m$ is called *overdetermined*, while the case $n < m$ is called *underdetermined*. In the case $n < m$, the matrix is singular (i.e. $N(A) \neq \{0\}$) and hence the function is not strongly convex. Nevertheless, the gradient descent features the following linear convergence rate

$$f(x^{(k)}) - \min f \leq (1 - \gamma\sigma_{\min}^2(2-\gamma L))^k (f(x^{(0)}) - \min f), \tag{4}$$

where $\sigma_{\min}$ is the minimum singular value of A.

(i) Implement the gradient descent algorithm for a general least square problem like (3).

(ii) Generate A randomly with Gaussian entries. Let $x_* \in \mathbb{R}^m$ and set $y = Ax_* + \varepsilon$, where ϵ is a Gaussian random vector with zero mean. Compute the minimum and maximum singular values of A and set the stepsize accordingly (consider also the choice $\gamma = 2/(L+\mu)$).

(iii) in the noiseless case ($\varepsilon = 0$), verify the theoretical bound (4) and compare it with the sublinear rate

$$f(x_k) - f(x_*) \leq \frac{1}{k}\frac{L}{2}\|x_0 - x_*\|^2, \qquad \text{valid for } \gamma = \frac{1}{L}.$$

(iv) Check the dependence on n and m.

3. Backtracking

In many situation it is difficult, if not impossible, to compute the Lipschitz constant L. In such cases a backtracking line search procedure will overcome the issue. Consider the least squares problem as in the previous point and determine the stepsize online. Let $\bar{\gamma} > 0$ and $\sigma \in \,]0,1[$. Then at each iteration k the stepsize in (1) is determined as follows

$$\begin{aligned}
\gamma_0 &= \bar{\gamma} \\
i_k &= \min\left\{ i \in \mathbb{N} \,\middle|\, f\big(x^{(k)} - \gamma_{k-1}\sigma^i \nabla f(x^{(k)})\big) \leq f(x_k) - \frac{1}{2}\gamma_{k-1}\sigma^i \big\|\nabla f(x^{(k)})\big\|^2 \right\} \\
\gamma_k &= \gamma_{k-1}\sigma^{i_k}
\end{aligned}$$

One can prove that the following rate of convergence holds

$$f(x_k) - \min f \leq \big(1 - \sigma\mu/L\big)^k \big(f(x^{(0)}) - \min f\big). \tag{5}$$

(i) check the computational limits of the svds function in Matlab for computing L in point (ii) of the previous exercise. (try $m = n = 10^4$).

(ii) Implement backtracking for the least square problem (3) and include it in the code.

(iii) Run the algorithm on large problems (e.g., $m = n = 10^4$) and for a limited number of total iterations (e.g., 20-30).

4. A curve fitting problem

Consider the following dictionary of polynomials

$$\varphi_j(x) = x^{j-1}, \quad j = 1, \ldots, m$$

and the function $f\colon \mathbb{R} \to \mathbb{R}$, $f(x) = \sin(\pi x)$. We assume that we have access only to a given data set $(x_i, y_i)_{1\leq i\leq n}$, where $x_i = -1 + 2(i-1)/(n-1)$ and $y_i = \sin(\pi x_i) + \varepsilon_i$ and ε_i is a zero mean Gaussian noise. The goal is to find the best approximation of f as $\sum_{j=1}^m \beta_j \varphi_j$ on $[-1, 1]$, based on the available data set. To that purpose we consider the least square approximation obtained by solving the following minimization problem

$$\min_{\beta\in\mathbb{R}^m} F(\beta) \qquad F(\beta) = \frac{1}{2}\sum_{i=1}^n \left(\sum_{j=1}^m \beta_j\varphi_j(x_i) - y_i\right)^2 = \frac{1}{2}\left\|A\beta - y\right\|^2, \tag{6}$$

where

$$A_{i,j} = \varphi_j(x_i) \qquad \text{and} \qquad y = (y_i)_{1\leq i\leq n}.$$

1. Implement the gradient descent method. Plot the solutions and compare with $\sin \pi x$ for $m = 20, 50, 100$ and $n = 10, 50, 100$. Try different values of the standard deviation of ε.

2. In case $\varepsilon = 0$, compare the coefficients of the solution with the coefficients in the Taylor expansion of $\sin \pi x$

$$\sin \pi x = \pi x - \frac{\pi^3}{3!}x^3 + \frac{\pi^5}{5!}x^5 - \frac{\pi^7}{7!}x^7 + \frac{\pi^9}{9!}x^9 - \cdots$$