Supplementary Material for Note 7

Convergence of Gradient Descent on Smooth Strongly-Convex Functions

Lecturer: Bin Hu, Date:02/17/2022

The convergence rate in Lecture Note 7 can be strengthened. We cover it here. Again, we focus on the performance of the gradient method for the unconstrained minimization problem

$$\min_{x \in \mathbb{R}^p} \; f(x) \tag{7.1}$$

where $f : \mathbb{R}^p \to \mathbb{R}$ is a differentiable function being L-smooth and m-strongly convex. We know there exists a unique global min x^* such that $f(x^*) \leq f(x)$ for all $x \in \mathbb{R}^p$. The gradient method iterates as follows

$$x_{k+1} = x_k - \alpha \nabla f(x_k) \tag{7.2}$$

The gradient method satisfies $\|x_k - x^*\| \leq \rho^k \|x_0 - x^*\|$ for some $0 < \rho < 1$ if a reasonable stepsize α is used. The smaller ρ is, the faster the gradient method converges to the optimal point x^*. However, ρ cannot be arbitrarily small (which means the gradient method cannot converge as fast as we want). Now let's try to understand how ρ depends on m, L, and α.

The main theorem describing how ρ depends on m, L, and α is stated as follows.

Theorem 7.1. *Suppose f is L-smooth and m-strongly convex. Let x^* be the unique global min. Given a stepsize α, if there exists $0 < \rho < 1$ and $\lambda \geq 0$ such that*

$$\begin{bmatrix} 1-\rho^2 & -\alpha \\ -\alpha & \alpha^2 \end{bmatrix} + \lambda \begin{bmatrix} -2mL & m+L \\ m+L & -2 \end{bmatrix} \tag{7.3}$$

is a negative semidefinite matrix, then the gradient method satisfies $\|x_k - x^*\| \leq \rho^k \|x_0 - x^*\|$.

The above theorem presents a sufficient testing condition for the linear convergence of the gradient method. We will use the theorem to analyze the convergence rate of the gradient method.

7.1 A Useful Lemma

Denote the $p \times p$ identity matrix as I. The following lemma is very helpful and will be used to prove Theorem 7.1.

Lemma 7.2. *Suppose the sequences* $\{\xi_k \in \mathbb{R}^p : k = 0, 1, \ldots\}$ *and* $\{u_k \in \mathbb{R}^p : k = 0, 1, 2, \ldots\}$ *satisfy* $\xi_{k+1} = \xi_k - \alpha u_k$. *In addition, assume the following inequality holds for all* k

$$\begin{bmatrix} \xi_k \\ u_k \end{bmatrix}^\mathsf{T} M \begin{bmatrix} \xi_k \\ u_k \end{bmatrix} \geq 0. \tag{7.4}$$

If there exist $0 < \rho < 1$ *and* $\lambda \geq 0$ *such that*

$$\begin{bmatrix} (1-\rho^2)I & -\alpha I \\ -\alpha I & \alpha^2 I \end{bmatrix} + \lambda M \tag{7.5}$$

is a negative semidefinite matrix, then the sequence $\{\xi_k : k = 0, 1, \ldots\}$ *satisfies* $\|\xi_k\| \leq \rho^k \|\xi_0\|$.

Proof: The key relation is

$$\|\xi_{k+1}\|^2 = \|\xi_k - \alpha u_k\|^2 = \|\xi_k\|^2 - 2\alpha(\xi_k)^\mathsf{T} u_k + \alpha^2 \|u_k\|^2 = \begin{bmatrix} \xi_k \\ u_k \end{bmatrix}^\mathsf{T} \begin{bmatrix} I & -\alpha I \\ -\alpha I & \alpha^2 I \end{bmatrix} \begin{bmatrix} \xi_k \\ u_k \end{bmatrix} \tag{7.6}$$

Since (7.5) is negative semidefinite, we have

$$\begin{bmatrix} \xi_k \\ u_k \end{bmatrix}^\mathsf{T} \left(\begin{bmatrix} (1-\rho^2)I & -\alpha I \\ -\alpha I & \alpha^2 I \end{bmatrix} + \lambda M \right) \begin{bmatrix} \xi_k \\ u_k \end{bmatrix} \leq 0 \tag{7.7}$$

We just expand the above inequality as

$$\begin{bmatrix} \xi_k \\ u_k \end{bmatrix}^\mathsf{T} \begin{bmatrix} I & -\alpha I \\ -\alpha I & \alpha^2 I \end{bmatrix} \begin{bmatrix} \xi_k \\ u_k \end{bmatrix} + \begin{bmatrix} \xi_k \\ u_k \end{bmatrix}^\mathsf{T} \begin{bmatrix} -\rho^2 I & 0_p \\ 0_p & 0_p \end{bmatrix} \begin{bmatrix} \xi_k \\ u_k \end{bmatrix} + \lambda \begin{bmatrix} \xi_k \\ u_k \end{bmatrix}^\mathsf{T} M \begin{bmatrix} \xi_k \\ u_k \end{bmatrix} \leq 0 \tag{7.8}$$

Applying the key relation (7.6), the above inequality can be rewritten as

$$\|\xi_{k+1}\|^2 - \rho^2 \|\xi_k\|^2 + \lambda \begin{bmatrix} \xi_k \\ u_k \end{bmatrix}^\mathsf{T} M \begin{bmatrix} \xi_k \\ u_k \end{bmatrix} \leq 0 \tag{7.9}$$

Due to the condition (7.4) and the non-negativity of λ, we have

$$\|\xi_{k+1}\|^2 - \rho^2 \|\xi_k\|^2 \leq -\lambda \begin{bmatrix} \xi_k \\ u_k \end{bmatrix}^\mathsf{T} M \begin{bmatrix} \xi_k \\ u_k \end{bmatrix} \leq 0$$

Hence $\|\xi_{k+1}\| \leq \rho \|\xi_k\|$ for all k. Therefore, we have $\|\xi_k\| \leq \rho \|\xi_{k-1}\| \leq \rho^2 \|\rho_{k-2}\| \leq \ldots \leq \rho^k \|\xi_0\|$. ■

It is emphasized that the condition (7.4) does not state that M is a positive semidefinite matrix. The inequality (7.4) is only assumed to hold for the two given sequences $\{\xi_k \in \mathbb{R}^p : k = 0, 1, \ldots\}$ and $\{u_k \in \mathbb{R}^p : k = 0, 1, 2, \ldots\}$. In addition, the relation $\xi_{k+1} = \xi_k - \alpha u_k$ is equivalent to

$$\xi_{k+1} = \begin{bmatrix} I & -\alpha I \end{bmatrix} \begin{bmatrix} \xi_k \\ u_k \end{bmatrix}$$

which states that ξ_{k+1} is a linear function of (ξ_k, u_k). This is the reason why $\|\xi_{k+1}\|^2$ is just a quadratic form of (ξ_k, u_k) as shown in (7.6).

7.2 Proof of Theorem 2.1

When f is L-smooth and m-strongly convex, one can prove the following inequality holds for $x, y \in \mathbb{R}^p$

$$(\nabla f(x) - \nabla f(y))^\mathsf{T}(x-y) \geq \frac{mL}{m+L}\|x-y\|^2 + \frac{1}{m+L}\|\nabla f(x) - \nabla f(y)\|^2 \tag{7.10}$$

This is the so-called co-coercivity property. You will be asked to prove this inequality in homework. This inequality can be rewritten as

$$\begin{bmatrix} x-y \\ \nabla f(x) - \nabla f(y) \end{bmatrix}^\mathsf{T} \begin{bmatrix} -2mLI & (m+L)I \\ (m+L)I & -2I \end{bmatrix} \begin{bmatrix} x-y \\ \nabla f(x) - \nabla f(y) \end{bmatrix} \geq 0. \tag{7.11}$$

Setting $y = x^*$ and noticing $\nabla f(x^*) = 0$, the above inequality leads to

$$\begin{bmatrix} x-x^* \\ \nabla f(x) \end{bmatrix}^\mathsf{T} \begin{bmatrix} -2mLI & (m+L)I \\ (m+L)I & -2I \end{bmatrix} \begin{bmatrix} x-x^* \\ \nabla f(x) \end{bmatrix} \geq 0. \tag{7.12}$$

The gradient method $x_{k+1} = x_k - \alpha\nabla f(x_k)$ can be rewritten as $x_{k+1} - x^* = x_k - x^* - \alpha\nabla f(x_k)$. We set $\xi_k = x_k - x^*$, and $u_k = \nabla f(x_k)$. Then the gradient method is exactly $\xi_{k+1} = \xi_k - \alpha u_k$ where (ξ_k, u_k) satisfies

$$\begin{bmatrix} \xi_k \\ u_k \end{bmatrix}^\mathsf{T} \begin{bmatrix} -2mLI & (m+L)I \\ (m+L)I & -2I \end{bmatrix} \begin{bmatrix} \xi_k \\ u_k \end{bmatrix} \geq 0. \tag{7.13}$$

The above inequality is just a restatement of (7.12). Therefore, we can choose $M = \begin{bmatrix} -2mLI & (m+L)I \\ (m+L)I & -2I \end{bmatrix}$ and apply Lemma 7.2 to directly prove Theorem 7.1. The final fact required for the proof is that $\begin{bmatrix} a & b \\ b & c \end{bmatrix}$ is negative semidefinite if and only if $\begin{bmatrix} aI & bI \\ bI & cI \end{bmatrix}$ is negative semidefinite (verify this!).

7.3 Convergence Rates of Gradient Method

Now we apply Theorem 7.1 to obtain the convergence rate ρ for the gradient method with various stepsize choices.

- Case 1: If we choose $\alpha = \frac{1}{L}$, $\rho = 1 - \frac{m}{L}$, and $\lambda = \frac{1}{L^2}$, we have

$$\begin{bmatrix} 1-\rho^2 & -\alpha \\ -\alpha & \alpha^2 \end{bmatrix} + \lambda \begin{bmatrix} -2mL & m+L \\ m+L & -2 \end{bmatrix} = \begin{bmatrix} -\frac{m^2}{L^2} & \frac{m}{L^2} \\ \frac{m}{L^2} & -\frac{1}{L^2} \end{bmatrix} = \frac{1}{L^2}\begin{bmatrix} -m^2 & m \\ m & -1 \end{bmatrix} \tag{7.14}$$

  The right side is clearly negative semidefinite due to the fact that $\begin{bmatrix} a \\ b \end{bmatrix}^\mathsf{T} \begin{bmatrix} -m^2 & m \\ m & -1 \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} = -(ma-b)^2 \leq 0$. Therefore, the gradient method with $\alpha = \frac{1}{L}$ converges as

$$\|x_k - x^*\| \leq \left(1 - \frac{m}{L}\right)^k \|x_0 - x^*\| \tag{7.15}$$

- Case 2: If we choose $\alpha = \frac{2}{m+L}$, $\rho = \frac{L-m}{L+m}$, and $\lambda = \frac{2}{(m+L)^2}$, we have

$$\begin{bmatrix} 1-\rho^2 & -\alpha \\ -\alpha & \alpha^2 \end{bmatrix} + \lambda \begin{bmatrix} -2mL & m+L \\ m+L & -2 \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \tag{7.16}$$

  The zero matrix is clearly negative semidefinite. Therefore, the gradient method with $\alpha = \frac{2}{m+L}$ converges as

$$\|x_k - x^*\| \leq \left(\frac{L-m}{L+m}\right)^k \|x_0 - x^*\| \tag{7.17}$$

Notice $L \geq m > 0$ and hence $1 - \frac{m}{L} \geq \frac{L-m}{L+m}$. This means the gradient method with $\alpha = \frac{2}{m+L}$ converges slightly faster than the case with $\alpha = \frac{1}{L}$. However, m is typically unknown in practice. The step choice of $\alpha = \frac{1}{L}$ is also more robust. The most popular choice for α is still $\frac{1}{L}$.

We can further express ρ as a function of α. To do this, we need to choose λ carefully for a given α. If we choose λ reasonably, we can show the best value for ρ that we can find is $\max\{|1-m\alpha|, |L\alpha - 1|\}$.

7.4 From convergence rate to iteration complexity

The convergence rate ρ naturally leads to an iteration number T guaranteeing the algorithm to achieve the so-called ε-optimality, i.e. $\|x_T - x^*\| \leq \varepsilon$ [1].

To guarantee $\|x_T - x^*\| \leq \varepsilon$, we can use the bound $\|x_T - x^*\| \leq \rho^T \|x_0 - x^*\|$. If we choose T such that $\rho^T \|x_0 - x^*\| \leq \varepsilon$, then we guarantee $\|x_T - x^*\| \leq \varepsilon$. Denote $c = \|x_0 - x^*\|$. Then $c\rho^k \leq \varepsilon$ is equivalent to

$$\log c + k \log \rho \leq \log(\varepsilon) \tag{7.18}$$

Notice $\rho < 1$ and $\log \rho < 0$. The above inequality is equivalent to

$$k \geq \log\left(\frac{\varepsilon}{c}\right) / \log \rho = \log\left(\frac{c}{\varepsilon}\right) / (-\log \rho) \tag{7.19}$$

So if we choose $T = \log\left(\frac{c}{\varepsilon}\right)/(-\log \rho)$, we guarantee $\|x_T - x^*\| \leq \varepsilon$.

Notice $\log \rho \leq \rho - 1 < 0$ (this can be proved using the concavity of log function), so $\frac{1}{1-\rho} \geq -\frac{1}{\log \rho}$ and we can also choose $T = \log\left(\frac{c}{\varepsilon}\right)/(1-\rho) \geq \log\left(\frac{c}{\varepsilon}\right)/(-\log \rho)$ to guarantee $\|x_T - x^*\| \leq \varepsilon$.

Another interpretation for $T = \log\left(\frac{c}{\varepsilon}\right)/(1-\rho)$ is that a first-order Taylor expansion of $-\log \rho$ at $\rho = 1$ leads to $-\log \rho \approx 1 - \rho$. So $\log\left(\frac{c}{\varepsilon}\right)/(-\log \rho)$ is roughly equal to $\log\left(\frac{c}{\varepsilon}\right)/(1-\rho)$ when ρ is close to 1.

[1] In many situations people require ε-optimal solution x_T to satisfy $f(x_T) - f(x^*) \leq \varepsilon$. We will talk about this case in late lectures. Typically this ends up with the same iteration complexity since we have $f(x) - f(x^*) = O(\|x - x^*\|^2)$ in many cases.

Clearly the smaller T is, the more efficient the optimization method is. The iteration number T describes the "ε-optimal iteration complexity" of the gradient method for smooth strongly-convex objective functions.

- For the gradient method with $\alpha = \frac{1}{L}$, we have $\rho = 1 - \frac{m}{L} = 1 - \frac{1}{\kappa}$ and hence $T = \log\left(\frac{c}{\varepsilon}\right)/(1-\rho) = \kappa \log\left(\frac{c}{\varepsilon}\right) = O\left(\kappa \log(\frac{1}{\varepsilon})\right)$. [2] Here we use the big O notation to highlight the dependence on κ and ε and hide the dependence on the constant c.

- For the gradient method with $\alpha = \frac{2}{L+m}$, we have $\rho = \frac{\kappa-1}{\kappa+1} = 1 - \frac{2}{\kappa+1}$ and hence $T = \log\left(\frac{c}{\varepsilon}\right)/(1-\rho) = \frac{\kappa+1}{2}\log\left(\frac{c}{\varepsilon}\right)$. Although $\frac{\kappa+1}{2} \leq \kappa$, we still have $\frac{\kappa+1}{2}\log\left(\frac{c}{\varepsilon}\right) = O\left(\kappa \log(\frac{1}{\varepsilon})\right)$. Therefore, the stepsize $\alpha = \frac{2}{m+L}$ can only improve the constant C hidden in the big O notation of the iteration complexity. People call this "improvement of a constant factor".

- In general, when ρ has the form $\rho = 1 - 1/(a\kappa + b)$, the resultant iteration complexity is always $O\left(\kappa \log(\frac{1}{\varepsilon})\right)$.

How shall we interpret the iteration complexity $O\left(\kappa \log(\frac{1}{\varepsilon})\right)$? It states that the required iteration T scales with the condition number κ. For larger κ, more iterations are required. This is consistent with our intuition since larger κ means the problem is ill-conditioned and more difficult to solve. There are algorithms which can significantly decrease the iteration complexity for unconstrained optimization problems with smooth strongly-convex objective functions. For example, Nesterov's method can decrease the iteration complexity from $O\left(\kappa \log(\frac{1}{\varepsilon})\right)$ to $O\left(\sqrt{\kappa} \log(\frac{1}{\varepsilon})\right)$. Momentum is used to accelerate optimization as:

$$x_{k+1} = x_k - \alpha \nabla f((1+\beta)x_k - \beta x_{k-1}) + \beta(x_k - x_{k-1}).$$

The theory for Nesterov's method is quite involved, and we skip those theoretical results here.

7.5 Two application examples

Finally we will discuss two application examples for unconstrained optimization with smooth strongly-convex objective functions.

7.5.1 Ridge regression

The ridge regression is formulated as an unconstrained minimization problem with the following objective function

$$f(x) = \frac{1}{n}\sum_{i=1}^{n}(a_i^\mathsf{T} x - b_i)^2 + \frac{\lambda}{2}\|x\|^2 \tag{7.20}$$

[2] For any functions $h(\varepsilon,\kappa)$ and $g(\varepsilon,\kappa)$, we say $h(\varepsilon,\kappa) = O(g(\varepsilon,\kappa))$ if there exists a constant C such that $|h(\varepsilon,\kappa)| \leq C|g(\varepsilon,\kappa)|$.

where $a_i \in \mathbb{R}^p$ and $b_i \in R$ are data points used to fit the linear model x.

- What is this problem about? The purpose of this problem is to fit a linear relationship between a and b. One wants to predict b from a as $b = a^\mathsf{T} x$. The ridge regression gives a way to find such x based on the observed pairs of (a_i, b_i).
- Why is there a term $\frac{\lambda}{2}\|x\|^2$? The term $\frac{\lambda}{2}\|x\|^2$ is called ℓ_2-regularizer. It confines the complexity of the linear predictors you want to use. The high-level idea is that you want x to work for all (a, b), not just the observed pairs (a_i, b_i). This is called "generalization" in machine learning. So adding such a term can induce the so-called stability and helps the predictor x to "generalize" for the data you have not seen. You need to take a machine learning course if you want to learn about generalization.
- What is λ? λ is a hyperparameter which is tuned to trade off training performance and generalization. For the purpose of this course, let's say λ is a fixed positive number. In practice, λ is typically set as a small number between 10^{-8} and 0.1.

This is a quadratic minimization problem with smooth strongly-convex objective functions, and the gradient method is guaranteed to achieve an iteration complexity of $O(\kappa \log(\frac{1}{\varepsilon}))$.

7.5.2 ℓ_2-Regularized Logistic regression

The ℓ_2-regularized logistic regression is formulated as an unconstrained minimization problem with the following objective function

$$f(x) = \frac{1}{n}\sum_{i=1}^{n} \log(1 + e^{-b_i a_i^\mathsf{T} x}) + \frac{\lambda}{2}\|x\|^2 \tag{7.21}$$

where $a_i \in \mathbb{R}^p$ and $b_i \in \{-1, 1\}$ are data points used to fit the linear model x.

- What is this problem about? The purpose of this problem is to fit a linear "**classifier**" between a and b. Let's say you have collected a lot of images for cats and dogs. You augment the pixels of any such image into a vector a and wants to predict whether the image is a cat or a dog. Let's say $b = 1$ if the image is a cat, and $b = -1$ if the image is a dog. So you want to predict b based on a. You want to find x such that $b = 1$ when $a^\mathsf{T} x \geq 0$, and $b = -1$ when $a^\mathsf{T} x < 0$. The logistic regression gives a way to find such x based on the observed feature/label pairs of (a_i, b_i). You may want to take a statistics course or a machine learning course if you want to learn more about logistic regression.
- Why is there a term $\frac{\lambda}{2}\|x\|^2$? Again, the term $\frac{\lambda}{2}\|x\|^2$ is the ℓ_2-regularizer. It is used to induce generalization and help x work on all the (a, b) not just the observed data points (a_i, b_i).

The function (7.21) is also L-smooth and m-strongly convex. Hence the gradient method can be applied here to achieve an iteration complexity of $O(\kappa \log(\frac{1}{\varepsilon}))$.