Introductory Lectures on Optimization

Descent Method (2)

Hui Qian
qianhui@zju.edu.cn

College of Computer Science, Zhejiang University

October 23, 2024

Outline

Part I
Newton Descent

Historical Origins

The Newton method is widely known as a technique for finding a root of a function of one variable. Let $\phi(t) : \mathbb{R} \to \mathbb{R}$. Consider the equation

$$\phi(t^*) = 0.$$

The Newton method is based on linear approximation. Assume that we get some t close enough to t^*. Note that

$$\phi(\underbrace{t + \Delta t}_{\text{expect that will be } t^*}) = \phi(t) + \phi'(t)\Delta t + o(|\Delta t|).$$

Historical Origins

Therefore the equation $\phi(t+\Delta t)=0$ can be approximated by the following linear equation:

$$\phi(t)+\phi'(t)\Delta t=0.$$

We can expect that the solution of this equation, the displacement Δt is a good approximation to the optimal displacement $\Delta t^*=t^*-t$. Converting this idea in an algorithmic form, we get the process

$$t_{k+1}=t_k-\frac{\phi(t_k)}{\phi'(t_k)}.$$

Historical Origins

This scheme can be naturally extended onto the problern of finding solution to a system of nonlinear equations

$$F(\boldsymbol{x}) = 0$$

where $\boldsymbol{x} \in \mathbb{R}^n$ and $F(\boldsymbol{x}) : \mathbb{R}^n \to \mathbb{R}^n$. In this case we have to define the displacernent $\Delta\boldsymbol{x}$ as a solution to the following system of linear equations:

$$F(\boldsymbol{x}) + J_F(\boldsymbol{x})\Delta\boldsymbol{x} = 0$$

(it is called the Newton system). If the Jacobian $J_F(\boldsymbol{x})$ is nondegenerate, we can compute displacernent $\Delta\boldsymbol{x} = -[J_F(\boldsymbol{x}_k)]^{-1}F(\boldsymbol{x}_k)$. The corresponding iterative scheme looks as follows:

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k - [J_F(\boldsymbol{x}_k)]^{-1}F(\boldsymbol{x}_k).$$

Basic Scheme

Finally, in view of the first order condition of continuously differentiable function , we can replace the unconstrained minimization problem by a problem of finding roots of the nonlinear system

$$\nabla f(\boldsymbol{x}) = 0. \tag{28}$$

Further, for solving (28) we can apply a standard Newton method for systems of nonlinear equations. In this case, the Newton system looks as follows:

$$\nabla f(\boldsymbol{x}) + \nabla^2 f(\boldsymbol{x}) \Delta \boldsymbol{x} = 0.$$

Hence, the Newton method for optimization problems appears to be in the form

$$\boxed{\boldsymbol{x}_{k+1} = \boldsymbol{x}_k - [\nabla^2 f(\boldsymbol{x}_k)]^{-1} \nabla f(\boldsymbol{x}_k).} \tag{29}$$

Basic Scheme

Note that we can also obtain the process (29), using the idea of quadratic approximation. Consider this approximation, computed with respect to the point $\boldsymbol{x}_k$:

$$\phi(\boldsymbol{x}) = f(\boldsymbol{x}_k) + \langle \nabla f(\boldsymbol{x}_k),\ \boldsymbol{x} - \boldsymbol{x}_k \rangle + \frac{1}{2} \langle \nabla^2 f(\boldsymbol{x}_k)(\boldsymbol{x} - \boldsymbol{x}_k),\ \boldsymbol{x} - \boldsymbol{x}_k \rangle.$$

Assume that $\nabla^2 f(\boldsymbol{x}_k) \succ 0$. Then we can choose $\boldsymbol{x}_{k+1}$ as a point of minimum of the quadratic function $\phi(\boldsymbol{x})$. This means that

$$\nabla \phi(\boldsymbol{x}_{k+1}) = \nabla f(\boldsymbol{x}_k) + \nabla^2 f(\boldsymbol{x}_k)(\boldsymbol{x}_{k+1} - \boldsymbol{x}_k) = 0,$$

and we come again to the Newton process (29)。

Basic Scheme

Remark.

The convergence of the Newton method in a neighborhood of a strict local minimum is very fast.

Two serious drawbacks.

- Firstly, it can break down if $\nabla^2 f(\boldsymbol{x}_k)$ is degenerate (a square matrix whose determinant is equal to zero).
- Secondly, the Newton process can diverge.

Basic Scheme

Example 7

Let us apply the Newton method for finding a root of the following function of one variable:

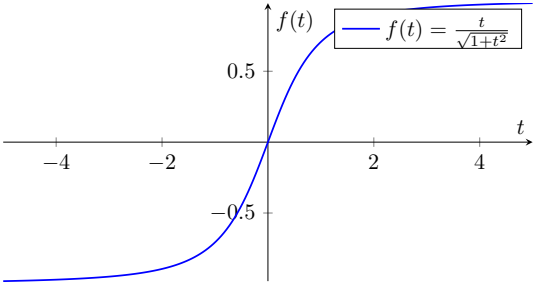
$$\phi(t) = \frac{t}{\sqrt{1+t^2}} \quad \text{and} \quad \phi'(t) = \frac{1}{[1+t^2]^{3/2}}.$$

Clearly, $t^* = 0$. Therefore the Newton process looks as follows:

$$t_{k+1} = t_k - \frac{\phi(t_k)}{\phi'(t_k)} = t_k - \frac{t_k}{\sqrt{1+t_k^2}} \cdot [1+t_k^2]^{3/2} = -t_k^3.$$

Thus, if $|t_0| < 1$, then this method converges and the convergence is extremely fast. The points ± 1 are the oscillation points of this method. If $|t_0| > 1$, then the method diverges.

Basic Scheme



$$\phi(t) = \frac{t}{\sqrt{1+t^2}}.$$

Damped Newton Method

In order to avoid a possible divergence, in practice we can apply a damped Newton method:

$$\boxed{\boldsymbol{x}_{k+1} = \boldsymbol{x}_k - h_k[\nabla^2 f(\boldsymbol{x}_k)]^{-1}\nabla f(\boldsymbol{x}_k),}$$

where, $h_k > 0$ is a step-size parameter.

Remark.

- At the initial stage of the method we can use the same step size strategies as for the gradient scheme.
- At the final stage it is reasonable to choose $h_k = 1$.

Local Convergence of The Newton Method.

Let us study the local convergence of the Newton method. Consider the problern

1. $f \in C_M^{2,2}(\mathbb{R}^n)$,
2. There exists a local minimum of function f with positive definite Hessian:

$$\nabla^2 f(\boldsymbol{x}^*) \succeq l I_n,\ l > 0. \tag{30}$$

3. Our starting point $\boldsymbol{x}_0$ is close enough to $\boldsymbol{x}^*$.

Consider the process:

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k - [\nabla^2 f(\boldsymbol{x}_k)]^{-1} \nabla f(\boldsymbol{x}_k).$$

Local Convergence of The Newton Method.

Using the same reasoning as for the gradient method, we obtain the following representation:

$$\begin{aligned}\boxed{\boldsymbol{x}_{k+1}-\boldsymbol{x}^*} &= \boldsymbol{x}_k-\boldsymbol{x}^*-[\nabla^2 f(\boldsymbol{x}_k)]^{-1}\nabla f(\boldsymbol{x}_k) \quad \text{by definition}\\ &= \boldsymbol{x}_k-\boldsymbol{x}^*-[\nabla^2 f(\boldsymbol{x}_k)]^{-1}\int_0^1 \nabla^2 f(\boldsymbol{x}^*+\tau(\boldsymbol{x}_k-\boldsymbol{x}^*))(\boldsymbol{x}_k-\boldsymbol{x}^*)d\tau\\ &\text{since we have } \nabla f(\boldsymbol{x}^*)=0\\ &= [\nabla^2 f(\boldsymbol{x}_k)]^{-1}G_k\cdot(\boxed{\boldsymbol{x}_k-\boldsymbol{x}^*}),\end{aligned} \tag{31}$$

where, $G_k=\int_0^1\left[\nabla^2 f(\boldsymbol{x}_k)-\nabla^2 f(\boldsymbol{x}^*+\tau(\boldsymbol{x}_k-\boldsymbol{x}^*))\right]d\tau$。

We consider the contracting mappings, thus both G_k and $[\nabla^2 f(\boldsymbol{x}_k)]^{-1}$ should be explored.

Bound for G_k

Denote $r_k = \|\boldsymbol{x}_k - \boldsymbol{x}^*\|$. Then

$$\begin{aligned}\|G_k\| &= \left\| \int_0^1 \left[\nabla^2 f(\boldsymbol{x}_k) - \nabla^2 f(\boldsymbol{x}^* + \tau(\boldsymbol{x}_k - \boldsymbol{x}^*))\right] d\tau \right\| \\ &\leq \int_0^1 \left\| \nabla^2 f(\boldsymbol{x}_k) - \nabla^2 f(\boldsymbol{x}^* + \tau(\boldsymbol{x}_k - \boldsymbol{x}^*)) \right\| d\tau \\ &\leq \int_0^1 M(1-\tau) r_k d\tau = \frac{r_k}{2} M. \quad C_M^{2,2}(\mathbb{R}^n) \text{ Lipschitz condition.}\end{aligned}$$

Bound for $[\nabla^2 f(x_k)]^{-1}$

In view of Corollary 1.2.2 of Nesterov [2003], we have for $f \in C_M^{2,2}(\mathbb{R}^n)$ and $\|\boldsymbol{y} - \boldsymbol{x}\| = r$,

$$\nabla^2 f(\boldsymbol{x}) - MrI_n \preceq \nabla^2 f(\boldsymbol{y}) \preceq \nabla^2 f(\boldsymbol{x}) + MrI_n,$$

where I_n is the identity matrix on $\mathbb{R}^n$.

In view of Corollary (1.2.2) and (30), we have

$$\nabla^2 f(\boldsymbol{x}_k) \succeq \nabla^2 f(\boldsymbol{x}^*) - Mr_k I_n \succeq (l - Mr_k) I_n.$$

Therefore, if $r_k < \frac{l}{M}$, then $\nabla^2 f(\boldsymbol{x}_k)$is positive definite and

$$\left\| [\nabla^2 f(\boldsymbol{x}_k)]^{-1} \right\| \leq (l - Mr_k)^{-1}.$$

Local Convergence of The Newton Method.

$$r_{k+1} = [\nabla^2 f(\boldsymbol{x}_k)]^{-1} G_k \cdot r_k,$$
$$\|G_k\| \leq \frac{r_k}{2} M,$$
$$\left\|[\nabla^2 f(\boldsymbol{x}_k)]^{-1}\right\| \leq (l - M r_k)^{-1}.$$

Hence, for r_k small enough ($r_k < \frac{2l}{3M}$), we have

$$r_{k+1} \leq \frac{M}{2(l - M r_k)} r_k^2 \ \ (< r_k).$$

The rate of convergence of this type is called quadratic.

Local Convergence of The Newton Method.

Thus, we have proved the following theorem.

Theorem 8

Let function $f(\boldsymbol{x})$ satisfy our assumptions. Suppose that the initial starting point $\boldsymbol{x}_0$ is close enough to $\boldsymbol{x}^*$:

$$\|\boldsymbol{x}_0 - \boldsymbol{x}^*\| < \bar{r} = \frac{2l}{3M}.$$

Then, $\|\boldsymbol{x}_k - \boldsymbol{x}^*\| < \bar{r}$ for all k, and the Newton method converges quadratically:

$$\|\boldsymbol{x}_{k+1} - \boldsymbol{x}^*\| \leq \frac{M \|\boldsymbol{x}_k - \boldsymbol{x}^*\|^2}{2(l - M \|\boldsymbol{x}_k - \boldsymbol{x}^*\|)}.$$

Other Approximation derived Methods: Variable Metric

Revisit GD: Let us fix some $\bar{x} \in \mathbb{R}^n$. Consider the following approximation of the function:

$$\phi_1(\boldsymbol{x}) = f(\bar{\boldsymbol{x}}) + \langle \nabla f(\bar{\boldsymbol{x}}),\ \boldsymbol{x} - \bar{\boldsymbol{x}} \rangle + \frac{1}{2h} \left\| \boldsymbol{x} - \bar{\boldsymbol{x}} \right\|^2,$$

where the parameter h is positive.
The first-order optimality condition provides us with the following equation for $\boldsymbol{x}_1^*$, the unconstrained minimum of the function $\phi_1(\boldsymbol{x})$:

$$\nabla \phi_1(\boldsymbol{x}_1^*) = \nabla f(\bar{\boldsymbol{x}}) + \frac{1}{h}(\boldsymbol{x}_1^* - \bar{\boldsymbol{x}}) = 0.$$

Thus, $\boldsymbol{x}_1^* = \bar{\boldsymbol{x}} - h \nabla f(\bar{\boldsymbol{x}})$. That is exactly the iterate of the gradient descent method.

Other Approximation derived Methods: Variable Metric

Revisit Newton: Further, consider a quadratic approximation of function $f(\boldsymbol{x})$:

$$\phi_2(\boldsymbol{x}) = f(\bar{\boldsymbol{x}}) + \langle \nabla f(\bar{\boldsymbol{x}}),\ \boldsymbol{x} - \bar{\boldsymbol{x}} \rangle + \frac{1}{2} \langle \nabla^2 f(\bar{\boldsymbol{x}})(\boldsymbol{x} - \bar{\boldsymbol{x}}),\ \boldsymbol{x} - \bar{\boldsymbol{x}} \rangle.$$

We have already seen that the minimum of this function is

$$\boldsymbol{x}_2^* = \bar{\boldsymbol{x}} - [\nabla^2 f(\bar{\boldsymbol{x}})]^{-1} \nabla f(\bar{\boldsymbol{x}}),$$

and that is exactly the iterate of the Newton method.

逆矩阵的计算复杂度大约是 $O(n^3)$。迭代次数虽少，但是每次迭代开销大。

Other Approximation derived Methods: Variable Metric

Lightweight Newton: Thus, we can try to use some approximations of function $f(\boldsymbol{x})$, which are better than $\phi_1(\boldsymbol{x})$ and which are less expensive than $\phi_2(\boldsymbol{x})$.

Let G be a positive definite $n \times n$ matrix. Denote

$$\phi_G(\boldsymbol{x}) = f(\bar{\boldsymbol{x}}) + \langle \nabla f(\bar{\boldsymbol{x}}),\ \boldsymbol{x} - \bar{\boldsymbol{x}} \rangle + \frac{1}{2} \langle G(\boldsymbol{x} - \bar{\boldsymbol{x}}),\ \boldsymbol{x} - \bar{\boldsymbol{x}} \rangle.$$

Its minimum we obtain from the above equation is

$$\boldsymbol{x}_G^* = \bar{\boldsymbol{x}} - G^{-1} \nabla f(\bar{\boldsymbol{x}}). \tag{32}$$

Other Approximation derived Methods: Variable Metric

Variable Metric: The first-order methods, which form a sequence of matrices

$$\{G_k\} : G_k \to \nabla^2 f(\boldsymbol{x}^*)$$

or

$$\{H_k\} : H_k \equiv G_k^{-1} \to [\nabla^2 f(\boldsymbol{x}^*)]^{-1},$$

are called the variable metric methods (sometime, use Quasi-Newton instead).

In these methods only the gradients are involved in the process of generating the sequeces $\{G_k\}$ or $\{H_k\}$.

Other Approximation derived Methods: Variable Metric

New Inner Product: Note that the gradient and the Hessian of a nonlinear function $f(\boldsymbol{x})$ are defined with respect to a standard Euclidean inner product on $\mathbb{R}^n$:

$$\langle \boldsymbol{x},\ \boldsymbol{y} \rangle = \sum_{i=1}^{n} \boldsymbol{x}^{(i)} \boldsymbol{y}^{(i)},\ \boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^n, \|\boldsymbol{x}\| = \langle \boldsymbol{x},\ \boldsymbol{x} \rangle^{1/2}.$$

Indeed, the definition of the gradient is

$$f(\boldsymbol{x} + h) = f(\boldsymbol{x}) + \langle \nabla f(\boldsymbol{x}),\ h \rangle + o(\|h\|),$$

and from this equation we derive its coordinate representation:

$$\nabla f(\boldsymbol{x}) = \left(\frac{\partial f(\boldsymbol{x})}{\partial \boldsymbol{x}^{(1)}}, \cdots, \frac{\partial f(\boldsymbol{x})}{\partial \boldsymbol{x}^{(n)}} \right)^{\top}.$$

Other Approximation derived Methods: Variable Metric

New Inner Product: Consider a symmetric positive definite $n \times n$-matrix A. For $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^n$ denote

$$\langle \boldsymbol{x},\ \boldsymbol{y} \rangle_A = \langle A\boldsymbol{x},\ \boldsymbol{y} \rangle, \quad \|\boldsymbol{x}\|_A = \langle A\boldsymbol{x},\ \boldsymbol{x} \rangle^{1/2}.$$

The function $\|\boldsymbol{x}\|_A$ is a new norm on $\mathbb{R}^n$. Note that topologically this new metric is equivalent to the old one:

$$\lambda_n(A)^{1/2} \|\boldsymbol{x}\| \leq \|\boldsymbol{x}\|_A \leq \lambda_1(A)^{1/2} \|\boldsymbol{x}\|,$$

where $\lambda_n(A)$ and $\lambda_1(A)$ are the smallest and largest eigenvalues of the matex A^*A respectively.

Other Approximation derived Methods: Variable Metric

New Inner Product: However, the gradient and the Hessian, computed with respect to the new inner product are changing:

$$\begin{aligned} f(\boldsymbol{x}+h) &= f(\boldsymbol{x}) + \langle \nabla f(\boldsymbol{x}),\ h\rangle + \frac{1}{2}\langle \nabla^2 f(\boldsymbol{x})h,\ h\rangle + o(\|h\|) \\ &= f(\boldsymbol{x}) + \langle A^{-1}\nabla f(\boldsymbol{x}),\ h\rangle_A + \frac{1}{2}\langle A^{-1}\nabla^2 f(\boldsymbol{x})h,\ h\rangle_A + o(\|h\|_A). \end{aligned}$$

Hence, $\nabla_A f(\boldsymbol{x}) = A^{-1}\nabla f(x)$ is the new gradient and $\nabla^2_A f(\boldsymbol{x}) = A^{-1}\nabla^2 f(\boldsymbol{x})$ is the new Hessian. Thus, the direction used in the Newton method $[\nabla^2 f(\boldsymbol{x})]^{-1}\nabla f(\boldsymbol{x})$ can be seen as a gradient computed with respect to the metric defined by $A = \nabla^2 f(\boldsymbol{x})$. Note that the Hessian of $f(\boldsymbol{x})$ at $\boldsymbol{x}$ computed with respect to $A = \nabla^2 f(\boldsymbol{x})$ is I_n.

Other Approximation derived Methods: Variable Metric

Example 9

Consider quadratic function

$$f(\boldsymbol{x}) = \alpha + \langle \boldsymbol{a},\ \boldsymbol{x} \rangle + \frac{1}{2}\langle A\boldsymbol{x},\ \boldsymbol{x} \rangle,$$

where $A = A^\top \succeq 0$. Note that $\nabla f(\boldsymbol{x}) = Ax + \boldsymbol{a}$, $\nabla^2 f(\boldsymbol{x}) = A$ and

$$\nabla f(\boldsymbol{x}^*) = A\boldsymbol{x}^* + \boldsymbol{a} = 0$$

for $\boldsymbol{x}^* = -A^{-1}\boldsymbol{a}$.

Other Approximation derived Methods: Variable Metric

Example 9 (Continued.) Let us compute the Newton direction at some $\boldsymbol{x} \in \mathbb{R}^n$:

$$d_N(\boldsymbol{x}) = [\nabla^2 f(\boldsymbol{x})]^{-1} \nabla f(\boldsymbol{x}) = A^{-1}(A\boldsymbol{x} + \boldsymbol{a}) = \boldsymbol{x} + A^{-1}\boldsymbol{a}.$$

Therefore for any $\boldsymbol{x} \in \mathbb{R}^n$ we have $\boldsymbol{x} - d_N(\boldsymbol{x}) = -A^{-1}\boldsymbol{a} = \boldsymbol{x}^*$. Thus, for a quadratic function the Newton method converges in one step. Note also that

$$\begin{aligned} f(\boldsymbol{x}) &= \alpha + \langle A^{-1}\boldsymbol{a},\, \boldsymbol{x}\rangle_A + \frac{1}{2}\|\boldsymbol{x}\|_A^2, \\ \nabla f_A(\boldsymbol{x}) &= A^{-1}\nabla f(\boldsymbol{x}) = d_N(\boldsymbol{x}), \\ \nabla^2 f_A(\boldsymbol{x}) &= A^{-1}\nabla^2 f(\boldsymbol{x}) = I_n. \end{aligned}$$

更一般的情况下，$A^{-1} = H$，A 可以不是 $\nabla^2 f$。

Other Approximation derived Methods: Variable Metric

Let us write down a general scheme of the variable metric methods.

Variable metric method

0. Choose $\boldsymbol{x}_0 \in \mathbb{R}^n$. Set $H_0 = I_0$.
   Compute $f(\boldsymbol{x}_0)$ and $\nabla f(\boldsymbol{x}_0)$.

1. kth iteration ($k \geq 0$).
   a. Set $p_k = H_k \nabla f(\boldsymbol{x}_k)$.
   b. Find $\boldsymbol{x}_{k+1} = \boldsymbol{x}_k - h_k p_k$.
   c. Compute $f(\boldsymbol{x}_{k+1})$ and $\nabla f(\boldsymbol{x}_{k+1})$.
   d. Update the matrix $H_k : H_k \to H_{k+1}$.

Other Approximation derived Methods: Variable Metric

Quasi-Newton rule: The variable metric schemes differ one from another only in implementation of Step 1d), which updates matrix H_k. For that, they use new information, accumulated at Step 1c), namely the gradient $\nabla f(\boldsymbol{x}_{k+1})$. The idea is justified by the following property of a quadratic function. Let

$$f(\boldsymbol{x}) = \alpha + \langle \alpha,\ \boldsymbol{x} \rangle + \frac{1}{2}\langle A\boldsymbol{x},\ \boldsymbol{x} \rangle,\ \nabla f(\boldsymbol{x}) = A\boldsymbol{x} + \alpha.$$

Then, for any $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^n$ we have $\nabla f(\boldsymbol{x}) - \nabla f(\boldsymbol{y}) = A(\boldsymbol{x} - \boldsymbol{y})$. This identity explains the origin of the so-called Quasi-Newton rule:

$$H_{k+1}\left(\nabla f(\boldsymbol{x}_{k+1}) - \nabla f(\boldsymbol{x}_k)\right) = \boldsymbol{x}_{k+1} - \boldsymbol{x}_k.$$

Other Approximation derived Methods: Variable Metric

Example 10

Denote

$$\Delta H_k = H_{k+1} - H_k, \quad \gamma_k = \nabla f(\boldsymbol{x}_{k+1}) - \nabla f(\boldsymbol{x}_k), \quad \delta_k = \boldsymbol{x}_{k+1} - \boldsymbol{x}_k.$$

Then the quasi-Newton relation is satisfied by the following rules.

1. Rank-one correcton scheme

$$\Delta H_k = \frac{(\delta_k - H_k\gamma_k)(\delta_k - H_k\gamma_k)^\top}{\langle \delta_k - H_k\gamma_k,\ \gamma_k \rangle}.$$

2. Davidon-Fletcher-Powell scheme (DFP))

$$\Delta H_k = \frac{\delta_k \delta_k^\top}{\langle \gamma_k,\ \delta_k \rangle} - \frac{H_k\gamma_k\gamma_k^\top H_k}{\langle H_k\gamma_k,\ \gamma_k \rangle}.$$

Other Approximation derived Methods: Variable Metric

Example 10

(Continued.)

3 Broyden-Fletcher-Goldfarb-Shanno scheme (BFGS)

$$\Delta H_k = \frac{H_k\gamma_k\delta_k^\top + \delta_k\gamma_k^\top H_k}{\langle H_k\gamma_k,\ \gamma_k\rangle} - \beta_k \frac{H_k\gamma_k\gamma_k^\top H_k}{\langle H_k\gamma_k,\ \gamma_k\rangle},$$

where $\beta_k = 1 + \langle\gamma_k,\ \delta_k\rangle/\langle H_k\gamma_k,\ \gamma_k\rangle$。

Clearly, there are many other possibilities. From the computational point of view, BFGS is considered as the most stable scheme.

Remarks on Variable Metric

1. Note that for quadratic functions the variable metric methods usually terminate in n iterations.
2. In a neighborhood of strict minimum they have a superlinear rate of convergence: for any $\boldsymbol{x}_0 \in \mathbb{R}^n$ there exists a number N such that for all $k \geq N$ we have

$$\|\boldsymbol{x}_{k+1} - \boldsymbol{x}^*\| \leq \text{const} \cdot \|\boldsymbol{x}_k - \boldsymbol{x}^*\| \cdot \|\boldsymbol{x}_{k-n} - \boldsymbol{x}^*\|$$

   (the proofs are very long and technical).
3. As far as global convergence is concerned, these methods are not better than the gradient method (at least, form the theoretical point of view).

Remarks on Variable Metric

4. Note that in the variable metric schemes it is necessary to store and update a symmetric $n \times n$-matrix. Thus, each iteration needs $O(n^2)$ auxiliary arithmetic operations. During many years this feature was considered as one of the main drawbacks of the variable metric methods. That stimulated the interest in so-called conjugate gradients schemes, which have much lower complexity of each iteration.

Part II
Conjugate Gradient

Methods Based on Krylov Subspace

In the previous sections, all $\boldsymbol{x}_k$ are spaned from $\text{Lin}\{\boldsymbol{x}_0, \nabla f(\boldsymbol{x}_i), i = 1, \cdots, k-1\}$. Essentially, pick a point from Lin. Is it possible to consider finding solutions in a subspace?

Krylov Subspace: To solve a large-scale sparse system of linear equations

$$A\boldsymbol{x} = \boldsymbol{b}, \quad A \in \mathbb{R}^{n\times n}, \boldsymbol{b} \in \mathbb{R}^n,$$

Krylov subspace can be used as a target subspace in many methods (Arnoldi, Lanczos, Conjugate gradient, GMRES).

Remark. This type of method is also regarded as a projection method, that is, to find the projection of the true solution in a certain subspace (which can be an orthogonal projection or an oblique projection).

Historical Origin of CG

The conjugate gradient method were initially proposed for the numerical solution of particular systems of linear equations (Stiefel [1952]). It also can be used to minimize quadratic functions like

$$\min_{\boldsymbol{x}\in\mathbb{R}^n} f(\boldsymbol{x}), \tag{33}$$

where $f(\boldsymbol{x}) = \alpha + \langle \boldsymbol{a},\ \boldsymbol{x}\rangle + \frac{1}{2}\langle A\boldsymbol{x},\ \boldsymbol{x}\rangle$, and $A = A^\top \succ 0$, since

$$A\boldsymbol{x} = -\boldsymbol{a}.$$

We have already seen that the solution to this problem is $\boxed{\boldsymbol{x}^* = -A^{-1}\boldsymbol{a}}$.

Historical Origin of CG

Therefore, our objective function can be written in the following form

$$\begin{aligned} f(\boldsymbol{x}) &= \alpha + \langle \boldsymbol{a},\ \boldsymbol{x} \rangle + \frac{1}{2}\langle A\boldsymbol{x},\ \boldsymbol{x} \rangle \\ &= \alpha - \langle A\boldsymbol{x}^*,\ \boldsymbol{x} \rangle + \frac{1}{2}\langle A\boldsymbol{x},\ \boldsymbol{x} \rangle \\ &= \alpha - \frac{1}{2}\langle A\boldsymbol{x}^*,\ \boldsymbol{x}^* \rangle + \frac{1}{2}\langle A(\boldsymbol{x} - \boldsymbol{x}^*),\ \boldsymbol{x} - \boldsymbol{x}^* \rangle. \end{aligned}$$

Thus,

$$f^* = \alpha - \frac{1}{2}\langle A\boldsymbol{x}^*,\ \boldsymbol{x}^* \rangle \text{ and}$$

$$\boxed{\nabla f(\boldsymbol{x}) = A(\boldsymbol{x} - \boldsymbol{x}^*)}.$$

Historical Origin of CG

Suppose we are given a starting point $\boldsymbol{x}_0 \in \mathbb{R}^n$. Consider the following linear Krylov subspace

$$\mathcal{L}_k = \text{Lin}\{A(\boldsymbol{x}_0 - \boldsymbol{x}^*), \ldots, A^k(\boldsymbol{x}_0 - \boldsymbol{x}^*)\},\ k \geq 1,$$

where A^k is the k-th power of matrix A. A sequence of points $\{\boldsymbol{x}_k\}$ is generated by the Conjugate Gradient Method in accorance with the folowing rule.

$$\boxed{\boldsymbol{x}_k = \text{argmin}\{f(\boldsymbol{x}) | \boldsymbol{x} \in \boldsymbol{x}_0 + \mathcal{L}_k\},\ k \geq 1.} \tag{34}$$

This definition looks quite artificial. However, later we will see that this method can be written in a pure "algorithmic"form. We need representation (34) only for theoretical analysis.

Historical Origin of CG

Calculating the optimum on $\boldsymbol{x}_0 + \mathcal{L}_k$ means that in the k-th iteration, k-th weights are to be calculated. This is not what we want. In practice, we hope to have the following calculation process.

- This is a descent algorithm. That is, for $k = 1, \ldots,$

$$\begin{aligned} \boldsymbol{x}_k = \boldsymbol{x}_{k-1} + \alpha_{k-1}\boldsymbol{p}_{k-1}, \Rightarrow (A\boldsymbol{x}_k + \boldsymbol{a}) &= (A\boldsymbol{x}_{k-1} + \boldsymbol{a}) + \alpha_{k-1}A\boldsymbol{p}_{k-1}, \\ \Rightarrow \boxed{\nabla f(\boldsymbol{x}_k) = \nabla f(\boldsymbol{x}_{k-1}) + \alpha_{k-1}A\boldsymbol{p}_{k-1}}&, \end{aligned} \tag{35}$$

  where $\alpha_{k-1} > 0$, for $k = 1, \ldots$.
- The direction is updated according to the gradient and last direction as follows. With $\beta_k > 0$, $p_0 = \nabla f(\boldsymbol{x}_0)$, and for $k = 1, \ldots,$

$$\boxed{\boldsymbol{p}_k = \nabla f(\boldsymbol{x}_k) - \beta_{k-1}\boldsymbol{p}_{k-1}}. \tag{36}$$

Fundamental Theory

Lemma 11

In view of (35) and (36), for any $k \geq 1$,

$$\begin{aligned} &\{\mathcal{L}_k \triangleq \text{Lin}\{A(\boldsymbol{x}_0 - \boldsymbol{x}^*), A^2(\boldsymbol{x}_0 - \boldsymbol{x}^*), \ldots, A^k(\boldsymbol{x}_0 - \boldsymbol{x}^*)\}\} \\ \equiv &\{\mathcal{R}_k \triangleq \text{Lin}\{\nabla f(\boldsymbol{x}_0), \nabla f(\boldsymbol{x}_1), \ldots, \nabla f(\boldsymbol{x}_{k-1})\}\} \end{aligned} \tag{37}$$

and

$$\begin{aligned} &\{\mathcal{L}_k \triangleq \text{Lin}\{A(\boldsymbol{x}_0 - \boldsymbol{x}^*), A^2(\boldsymbol{x}_0 - x^*), \ldots, A^k(\boldsymbol{x}_0 - \boldsymbol{x}^*)\}\} \\ \equiv &\{\mathcal{P}_k \triangleq \text{Lin}\{\boldsymbol{p}_0, \boldsymbol{p}_1, \ldots, \boldsymbol{p}_{k-1}\}\}. \end{aligned} \tag{38}$$

Fundamental Theory

Proof. The proof is by induction. (37) and (38) hold trivially for $k = 1$. since

$$\mathcal{L}_1 = \mathcal{R}_1 = \mathcal{P}_1, \text{ since } \nabla f(\boldsymbol{x}_0) = A(\boldsymbol{x}_0 - \boldsymbol{x}^*) = \boldsymbol{p}_0.$$

We here suppose that

$$\mathcal{L}_k = \mathcal{R}_k \text{ and } \mathcal{L}_k = \mathcal{P}_k. \tag{39}$$

Let's check the $\nabla f(\boldsymbol{x}_k)$ in $\mathcal{R}_{k+1}$ and $A^{k+1}(\boldsymbol{x}_0 - \boldsymbol{x}^*)$ in $\mathcal{L}_{k+1}$.

- For $\nabla f(\boldsymbol{x}_k)$: In view of (35), we have

$$\begin{aligned} \nabla f(\boldsymbol{x}_k) &= \nabla f(\boldsymbol{x}_{k-1}) + \alpha_{k-1} A \boldsymbol{p}_{k-1} \\ &\in \mathcal{R}_k \cup A\mathcal{P}_k \\ &= \mathcal{L}_k \cup A\mathcal{L}_k = \mathcal{L}_{k+1}. \end{aligned}$$

Fundamental Theory

Proof. (Continued.)

- For $A^{k+1}(\boldsymbol{x}_0 - \boldsymbol{x}^*)$: In view of (39), we have

$$\begin{aligned} A^{k+1}(\boldsymbol{x}_0 - \boldsymbol{x}^*) &= A(A^k(\boldsymbol{x}_0 - \boldsymbol{x}^*)) \in A\mathcal{L}_k = A\mathcal{P}_k \\ &= \text{Lin}\{A\boldsymbol{p}_0, A\boldsymbol{p}_1, \ldots, A\boldsymbol{p}_{k-1}\} = \text{Lin}\{\nabla f(\boldsymbol{x}_0), \nabla f(\boldsymbol{x}_1), \ldots, \nabla f(\boldsymbol{x}_k)\}. \end{aligned}$$

The last equality comes from the fact that in view of (35), we have

$$A\boldsymbol{p}_0 = \frac{1}{\alpha_0}(\nabla f(\boldsymbol{x}_1) - \nabla f(\boldsymbol{x}_0)), \ldots, A\boldsymbol{p}_{k-1} = \frac{1}{\alpha_{k-1}}(\nabla f(\boldsymbol{x}_k) - \nabla f(\boldsymbol{x}_{k-1})).$$

Thus, $A^{k+1}(\boldsymbol{x}_0 - \boldsymbol{x}^*) \in \mathcal{R}_{k+1}$. Thus, we arrive at $\mathcal{R}_{k+1} = \mathcal{L}_{k+1}$.

□

Fundamental Theory

CG: a two-term recurrence

$$\begin{aligned} \boldsymbol{x}_k &= \boldsymbol{x}_{k-1} + \alpha_{k-1}\boldsymbol{p}_{k-1}, \\ \boldsymbol{p}_k &= \nabla f(\boldsymbol{x}_k) - \beta_{k-1}\boldsymbol{p}_{k-1}. \end{aligned}$$

Thus, we have

$$\begin{aligned} \boldsymbol{x}_{k+1} &= \boldsymbol{x}_k + \alpha_k \boldsymbol{p}_k \\ &= \boldsymbol{x}_k + \alpha_k \left(\nabla f(\boldsymbol{x}_k) - \beta_{k-1}\boldsymbol{p}_{k-1}\right) \\ &= \boldsymbol{x}_k + \alpha_k \nabla f(\boldsymbol{x}_k) - \underbrace{\alpha_k \beta_{k-1}\boldsymbol{p}_{k-1}}_{\in \mathcal{L}_k}. \end{aligned} \tag{40}$$

Fundamental Theory

Lemma 12 (Lemma 1.3.2)
For any $k, i \geq 0, k \neq i$, we have $\langle \nabla f(\boldsymbol{x}_k),\ \nabla f(\boldsymbol{x}_i) \rangle = 0$.

Proof. Let $k > i$. Consider the function

$$\phi(\lambda) = f\left(\boldsymbol{x}_0 + \sum_{j=1}^{k} \lambda^{(j)} \nabla f(\boldsymbol{x}_{j-1})\right),\ \lambda \in \mathbb{R}^k.$$

In view of Lemma11, for some λ_* we have $\boldsymbol{x}_k = \boldsymbol{x}_0 + \sum_{j=1}^{k} \lambda_*^{(j)} \nabla f(\boldsymbol{x}_{j-1})$.

Fundamental Theory

Lemma 12
For any $k, i \geq 0, k \neq i$, we have $\langle \nabla f(\boldsymbol{x}_k),\ \nabla f(\boldsymbol{x}_i) \rangle = 0$.

Proof. (Continued)
However, by definition, $\boldsymbol{x}_k$ is the minimum point of $f(\boldsymbol{x})$ on $\boldsymbol{x}_0 + \mathcal{L}_k$. Therefore $\nabla \phi(\lambda_*) = 0$. It remains to compute the components of the gradient:

$$0 = \frac{\partial \phi(\lambda_*)}{\partial \lambda^{(i)}} = \langle \frac{\partial \phi(\lambda_*)}{\partial \boldsymbol{x}_k},\ \frac{\partial \boldsymbol{x}_k}{\partial \lambda^{(i)}} \rangle = \langle \nabla f(\boldsymbol{x}_k),\ \nabla f(\boldsymbol{x}_i) \rangle.$$

□

Fundamental Theory

Corollary 13

The sequence generated by the Conjugate Gradient Method for problem (33) is finite.

证明.

Indeed, the number of nonzero orthogonal directions in $\mathbb{R}^n$ cannot exceed n. □

Corollary 14

For any $p \in \mathcal{L}_k$, $k \geq 1$, we have $\langle \nabla f(x_k),\ p \rangle = 0$。

$\mathcal{L}_k = \mathcal{P}_k$, $\nabla f(\boldsymbol{x}_k) \in \mathcal{L}_{k+1}$ and $\nabla f(\boldsymbol{x}_k) \notin \mathcal{L}_k$.

Fundamental Theory

The last auxiliary result explains the name of the method. Let $\delta_i = \boldsymbol{x}_{i+1} - \boldsymbol{x}_i$. It is clear that $\mathcal{L}_k = \text{Lin}\{\delta_0, \ldots, \delta_{k-1}\}$. (since we have $\delta_i = c\boldsymbol{p}_i$)

Lemma 15
For any $k \neq i$, we have $\langle A\delta_k,\ \delta_i \rangle = 0$.

(Such directions are called conjugate with respect to A)

Proof. Without loss of generality, we can assume that $k > i$. Then (Note that $\nabla f(\boldsymbol{x}) = A(\boldsymbol{x} - \boldsymbol{x}^*)$

$$\langle A\delta_k,\ \delta_i \rangle = \langle A(\boldsymbol{x}_{k+1} - \boldsymbol{x}_k),\ \delta_i \rangle = \langle \underbrace{\nabla f(\boldsymbol{x}_{k+1})}_{k+2 \text{ Kry}} - \underbrace{\nabla f(\boldsymbol{x}_k)}_{k+1 \text{ Kry}},\ \delta_i \rangle = 0,$$

since $\delta_i = \boldsymbol{x}_{i+1} - \boldsymbol{x}_i \in \mathcal{L}_{i+1} \subseteq \mathcal{L}_k$. □

CG Algorithm

Let us show how we can write down the Conjugate Gradient Method in a more algorithmic form. Since $\mathcal{L}_{k+1} = \text{Lin}\{\delta_0, \ldots, \delta_k\}$, accorrding to (40), we can represent $\boldsymbol{x}_{k+1}$ as follows:

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k - h_k \nabla f(\boldsymbol{x}_k) + \sum_{j=0}^{k-1} \lambda^{(j)} \delta_j.$$

In our notation, this is

$$\delta_k = -h_k \nabla f(\boldsymbol{x}_k) + \sum_{j=0}^{k-1} \lambda^{(j)} \delta_j. \tag{41}$$

$$\begin{aligned} \mathcal{L}_{k+1} &= \text{Lin}\{\delta_0, \ldots, \delta_k\} = \text{Lin}\{\nabla f(\boldsymbol{x}_0), \ldots, \nabla f(\boldsymbol{x}_k)\} \\ &= \text{Lin}\{\delta_0, \ldots, \delta_{k-1}, \nabla f(\boldsymbol{x}_k)\} \end{aligned}$$

CG Algorithm

Let us compute the coefficients in this representation. Multiplying (41) by A and $\delta_i, 0 \leq i \leq k-1$, and using lemma 15, we obtain

$$
\begin{aligned}
0 = \langle A\delta_k,\ \delta_i\rangle &= -h_k\langle A\nabla f(\boldsymbol{x}_k),\ \delta_i\rangle + \sum_{j=0}^{k-1} \lambda^{(j)}\langle A\delta_j,\ \delta_i\rangle \\
&= -h_k\langle A\nabla f(\boldsymbol{x}_k),\ \delta_i\rangle + \lambda^{(i)}\langle A\delta_i,\ \delta_i\rangle \\
&= \underbrace{-h_k\langle \nabla f(\boldsymbol{x}_k),\ \nabla f(\boldsymbol{x}_{i+1}) - \nabla f(\boldsymbol{x}_i)\rangle}_{i<k-1 \Rightarrow \text{above}=0} + \underbrace{\lambda^{(i)}\langle A\delta_i,\ \delta_i\rangle}_{i<k-1 \Rightarrow \lambda^{(i)}=0 \text{ since } \langle A\delta_i,\ \delta_i\rangle \text{ is positive.}} .
\end{aligned}
$$

CG Algorithm

Hence, in view of Lemma 12, $\lambda^{(i)} = 0$, for $i < k-1$. For $\boxed{i = k-1}$, we have

$$\lambda^{k-1} = \frac{h_k \left\|\nabla f(\boldsymbol{x}_k)\right\|^2}{\langle A\delta_{k-1},\ \delta_{k-1}\rangle} = \frac{h_k \left\|\nabla f(\boldsymbol{x}_k)\right\|^2}{\langle \nabla f(\boldsymbol{x}_k) - \nabla f(\boldsymbol{x}_{k-1}),\ \delta_{k-1}\rangle}.$$

Thus, $\boldsymbol{x}_{k+1} = \boldsymbol{x}_k - h_k\left(\nabla f(\boldsymbol{x}_k) - \lambda^{(k-1)}\delta_{k-1}\right) = \boldsymbol{x}_k - h_k \boldsymbol{p}_k$, where

$$\begin{aligned}\boldsymbol{p}_k &= \nabla f(\boldsymbol{x}_k) - \frac{\left\|\nabla f(\boldsymbol{x}_k)\right\|^2 \delta_{k-1}}{\langle \nabla f(\boldsymbol{x}_k) - \nabla f(\boldsymbol{x}_{k-1}),\ \delta_{k-1}\rangle} \\ &= \nabla f(\boldsymbol{x}_k) - \frac{\left\|\nabla f(\boldsymbol{x}_k)\right\|^2 \boldsymbol{p}_{k-1}}{\langle \nabla f(\boldsymbol{x}_k) - \nabla f(\boldsymbol{x}_{k-1}),\ \boldsymbol{p}_{k-1}\rangle},\end{aligned}$$

since $\delta_{k-1} = \boldsymbol{x}_k - \boldsymbol{x}_{k-1} = -h_{k-1}\boldsymbol{p}_{k-1}$ by definition of the directions $\{\boldsymbol{p}_k\}$.

CG Algorithm

Conjugate Gradient Method

0. Let $\boldsymbol{x}_0 \in \mathbb{R}^n$. Compute $f(\boldsymbol{x}_0)$, $\nabla f(\boldsymbol{x}_0)$。设 $\boldsymbol{p}_0 = \nabla f(\boldsymbol{x}_0)$。

1. k-th iteration ($k \geq 0$)
   - a Find $\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + h_k \boldsymbol{p}_k$ (by "exact" line search).
   - b Compute $f(\boldsymbol{x}_{k+1})$ and $\nabla f(\boldsymbol{x}_{k+1})$.
   - c Compute the coefficient β_k.
   - d Define $\boldsymbol{p}_{k+1} = \nabla f(\boldsymbol{x}_{k+1}) - \beta_k \boldsymbol{p}_k$.

CG Algorithm

The specification of the coefficient β_k.

1. Dai-Yuan

$$\beta_k = \frac{\left\|\nabla f(\boldsymbol{x}_{(k+1)})\right\|^2}{\langle \nabla f(\boldsymbol{x}_{k+1}) - \nabla f(\boldsymbol{x}_k),\ \boldsymbol{p}_k \rangle}.$$

2. Fletcher-Rieves:

$$\beta_k = -\frac{\left\|\nabla f(\boldsymbol{x}_{(k+1)})\right\|^2}{\left\|\nabla f(\boldsymbol{x}_{(k)})\right\|^2}.$$

3. Polak-Ribbiere:

$$\beta_k = -\frac{\langle \nabla f(\boldsymbol{x}_{k+1}),\ \nabla f(\boldsymbol{x}_{k+1}) - \nabla f(\boldsymbol{x}_k) \rangle}{\left\|\nabla f(\boldsymbol{x}_{(k)})\right\|^2}.$$

CG Algorithm

- Recall that in the quadratic case, the Conjugate Gradient Method terminates in n iterations (or less). Algorithmically, this means that $\boldsymbol{p}_n = 0$.
- In the general nonlinear case, this is not true. However, after n iterations, this direction loses its interpretation. Therefore, in all practical schemes, there exists a restarting strategy. This ensures the global convergence of the process.

In a neighborhood of a strict minimum, the conjugate gradient schemes demonstrate a local n-step quadratic convergence:

$$\|\boldsymbol{x}_{n+1} - \boldsymbol{x}^*\| \leq \text{const} \cdot \|\boldsymbol{x}_0 - \boldsymbol{x}^*\|^2$$

Note that this local convergence is slower than that of the variable metric methods. However, the conjugate gradient methods have the advantage of cheap iteration. As far as the global convergence is concerned, these schemes, in general, are not better than the simplest Gradient Method.

References I

Yurii Nesterov. *Introductory Lectures on Convex Optimization: A Basic Course*, volume 87. Springer Science & Business Media, 2003.

Eduard Stiefel. Methods of conjugate gradients for solving linear systems. *J. Res. Nat. Bur. Standards*, 49:409–435, 1952.

Lloyd N Trefethen and David Bau III. *Numerical linear algebra*, volume 50. Siam, 1997.

Stephen Wright, Jorge Nocedal, et al. Numerical optimization. *Springer Science*, 35(67-68): 7, 1999.

Stephen J Wright and Benjamin Recht. *Optimization for data analysis*. Cambridge University Press, 2022.

Stephen Boyd, Stephen P Boyd, and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

Yurii Nesterov. *Lectures on convex optimization*, volume 137. Springer, 2018.

Thank You!

Email:qianhui@zju.edu.cn