

ISEN 629: Engineering Optimization

Lecture 4

Sergiy Butenko

Industrial and Systems Engineering
Texas A& M University

Fall 2007

1/18

Newton's method: Guaranteeing the descent

Consider an iteration of the Newton's method:

$$x_{k+1} = x_k - (\nabla_k^2)^{-1} \nabla_k.$$

Here we assume that $\nabla_k \neq 0$.

- ▶ In general, the Newton's method may not possess the descent property, i.e. we may have $f(x_{k+1}) \geq f(x_k)$.
- ▶ However, if ∇_k^2 is positive definite, the Newton's direction $d_k = -(\nabla_k^2)^{-1} \nabla_k$ is a descent direction in the sense that there exists $\bar{\alpha}$ such that for all $0 < \alpha < \bar{\alpha}$: $f(x_k + \alpha d_k) < f(x_k)$.
- ▶ Therefore, we can modify the Newton's method to enforce the descent property as follows:
 - ▶ Find $\alpha_k = \arg \min_{\alpha \geq 0} (f(x_k - \alpha (\nabla_k^2)^{-1} \nabla_k))$
 - ▶ Set $x_{k+1} = x_k - \alpha_k (\nabla_k^2)^{-1} \nabla_k$.

2/18

Levenberg-Marquardt modification

If the Hessian is not positive definite, it can still be modified so that the iteration resulting from this modification will have the descent property.

Consider the matrix

$$M_k = \nabla_k^2 + \mu_k I_n,$$

where I_n is the $n \times n$ identity matrix. Note that if we denote by $\lambda_i, i = 1, \dots, n$ the eigenvalues of ∇_k^2 , then the eigenvalues of M_k are given by

$$\lambda_i + \mu_k, \quad i = 1, \dots, n.$$

Indeed, if v_i is the eigenvector of ∇_k^2 corresponding to the eigenvalue λ_i , then

$$M_k v_i = (\nabla_k^2 + \mu_k I_n) v_i = \lambda_i v_i + \mu_k v_i = (\lambda_i + \mu_k) v_i.$$

Hence, if we choose $\mu_k > |\lambda_{\min}(\nabla_k^2)|$, where $\lambda_{\min}(\nabla_k^2)$ is the minimum eigenvalue of ∇_k^2 , then all eigenvalues of M_k are positive, so M_k is a positive definite matrix.

3/18

Levenberg-Marquardt modification

To make sure that the descent property holds, we can use the direction $-M_k^{-1} \nabla_k$ instead of the direction $-(\nabla_k^2)^{-1} \nabla_k$ used in the Newton's method. Including the step size, we obtain the following iteration:

$$x_{k+1} = x_k - \alpha_k M_k^{-1} \nabla_k,$$

where $\alpha_k = \arg \min_{\alpha \geq 0} f(x_k - \alpha M_k^{-1} \nabla_k)$. This method is referred to as the *Levenberg-Marquardt modification*. With this modification of the Hessian matrix, the direction used becomes a descent direction. To see this, we write down the derivative of the function $\phi_k(\alpha) = f(x_k - \alpha M_k^{-1} \nabla_k)$. We have

$$\phi_k'(0) = -\nabla_k^T (M_k)^{-1} \nabla_k < 0,$$

since M_k^{-1} is positive definite.

4/18

Levenberg-Marquardt modification

The Levenberg-Marquardt modification is in some sense intermediate between the steepest descent and the Newton's method. If $\mu_k = 0$, then it coincides with the Newton's method. On the other hand, if μ_k is very large, then $M_k \approx C I_n$ for some very large $C > 0$, so $M_k^{-1} \approx \epsilon I_n$ for some small $\epsilon = 1/C > 0$, and the iteration is

$$x_{k+1} \approx x_k - \alpha_k \epsilon \nabla_k.$$

Thus, we obtain an approximation of the steepest descent iteration.

5/18

Quasi-Newton Methods

Recall a step of the Newton's method with step size:

$$x_{k+1} = x_k - \alpha_k (\nabla_k^2)^{-1} \nabla_k, \quad k \geq 0.$$

Quasi-Newton methods are modifications of the Newton's method in which at each step the inverse of the Hessian is approximated with a matrix that does not involve second-order derivatives. Thus, if we replace $(\nabla_k^2)^{-1}$ with its approximation H_k , we obtain a step of a quasi-Newton method:

$$x_{k+1} = x_k - \alpha_k H_k \nabla_k, \quad k \geq 0.$$

Different choices of H_k result in different variations of the quasi-Newton methods. Matrix H_k will be chosen so that it satisfies some properties that the Hessian matrix satisfies in the case of a convex quadratic function $f(x) = \frac{1}{2} x^T Q x + c^T x$.

6/18

Quasi-Newton Methods

In this case, since $\nabla_k = Q x_k + c$, we have

$$Q(x_{k+1} - x_k) = \nabla_{k+1} - \nabla_k.$$

If we denote by $p_k = x_{k+1} - x_k$ and by $g_k = \nabla_{k+1} - \nabla_k$, then

$$Q p_k = g_k, \quad k \geq 0 \quad \Leftrightarrow \quad p_k = Q^{-1} g_k, \quad k \geq 0.$$

Therefore, we choose H_k such that

$$p_i = H_k g_i, \quad i = 0, \dots, k-1, \text{ so}$$

after $n-1$ steps we obtain

$$H_n g_i = p_i, \quad i = 0, \dots, n-1.$$

This can be written in the matrix form as

$$H_n [g_0 \ g_1 \ \dots \ g_{n-1}] = [p_0 \ p_1 \ \dots \ p_{n-1}],$$

or, if we denote by $G_n = [g_0 \ g_1 \ \dots \ g_{n-1}]$ and by $P_n = [p_0 \ p_1 \ \dots \ p_{n-1}]$, then

$$H_n G_n = P_n.$$

7/18

Quasi-Newton Methods

If G_n is a nonsingular matrix, we have

$$H_n = P_n G_n^{-1}.$$

Similarly, we can show that

$$Q^{-1} G_n = P_n$$

and

$$Q^{-1} = P_n G_n^{-1}.$$

So, for the quadratic function, $H_n = Q^{-1}$. This means that after $n+1$ steps of a quasi-Newton method we get the same answer as we get after one step of the Newton's method, which is the global minimizer of the convex quadratic function. In fact, we can show that the global minimizer of the convex quadratic function is obtained in no more than n steps of a quasi-Newton method.

8/18

Rank-one correction formula

In rank one correction formula, we start with a symmetric positive definite matrix H_0 (say, $H_0 = I_n$). At step $k + 1$ we add a rank-one matrix to H_k to obtain H_{k+1} :

$$H_{k+1} = H_k + a_k z_k z_k^T \quad (1)$$

for some vector z_k and a scalar a_k (note that $z_k z_k^T$ is a $n \times n$ matrix of rank 1). As before, we require that

$$H_{k+1} g_i = p_i, \quad i = 0, \dots, k.$$

After some algebraic manipulations, we obtain the following rank one correction formula:

$$H_{k+1} = H_k + \frac{(p_k - H_k g_k)(p_k - H_k g_k)^T}{g_k^T (p_k - H_k g_k)}.$$

9/18

Other correction formulas

One of the drawbacks of this formula is that given a positive definite H_k , the resulting matrix H_{k+1} is not guaranteed to be positive definite. Some other quasi-Newton methods do guarantee such a property. For example, one of the most popular classes of quasi-Newton methods uses the following correction formula:

$$H_{k+1} = H_k + \frac{p_k p_k^T}{p_k^T g_k} - \frac{H_k g_k g_k^T H_k}{g_k^T H_k g_k} + \xi_k g_k^T H_k g_k \left(\frac{p_k}{p_k^T g_k} - \frac{H_k g_k}{g_k^T H_k g_k} \right) \left(\frac{p_k}{p_k^T g_k} - \frac{H_k g_k}{g_k^T H_k g_k} \right)^T,$$

where $p_k = x_{k+1} - x_k$, $g_k = \nabla_{k+1} - \nabla_k$, the parameters ξ_k satisfy

$$0 \leq \xi_k \leq 1$$

for all k , and H_0 is an arbitrary positive definite matrix. The scalars ξ_k parameterize the method (different values of ξ_k yield different algorithms).

10/18

Other correction formulas

$$H_{k+1} = H_k + \frac{p_k p_k^T}{p_k^T g_k} - \frac{H_k g_k g_k^T H_k}{g_k^T H_k g_k} + \xi_k g_k^T H_k g_k \left(\frac{p_k}{p_k^T g_k} - \frac{H_k g_k}{g_k^T H_k g_k} \right) \left(\frac{p_k}{p_k^T g_k} - \frac{H_k g_k}{g_k^T H_k g_k} \right)^T,$$

- ▶ If $\xi_k = 0$ for all k , we obtain the *Davidon-Fletcher-Powell* (DFP) method (historically the first quasi-Newton method).
- ▶ If $\xi_k = 1$ for all k , we obtain the *Broyden-Fletcher-Goldfarb-Shanno* (BFGS) method (considered one of the best general purpose quasi-Newton method).

11/18

Rate of convergence

- ▶ For quadratic functions, the quasi-Newton methods usually terminate in n iterations.
- ▶ In a neighborhood of strict minimum they have a superlinear rate of convergence: for any $x_0 \in \mathbb{R}^n$ there exists a number N such that for all $k \geq N$ we have

$$\|x_{k+1} - x_k\| \leq C \|x_k - x^*\| \cdot \|x_{k-n} - x^*\|.$$

- ▶ Like with the gradient methods, we have a global convergence to a stationary point (does not have to be a local minimizer).

12/18

Conjugate gradients

- ▶ Note that in quasi-Newton schemes we need to store and update a symmetric $n \times n$ -matrix.
- ▶ Thus, each iteration needs $O(n^2)$ auxiliary arithmetic operations.
- ▶ This feature was considered one of the main drawbacks of the quasi-Newton methods (with the modern computing capabilities, this is not that important anymore).
- ▶ This stimulated the development of conjugate gradient methods, which have much lower per-iteration complexity.

13/18

Conjugate directions

Given a positive definite $n \times n$ matrix Q , the nonzero directions d_0, d_1, \dots, d_k are called Q -conjugate if

$$d_i^T Q d_j = 0 \quad \text{for } i \neq j.$$

We first show that Q -conjugate directions d_0, d_1, \dots, d_k form a set of linearly independent vectors. Consider a linear combination of the given Q -conjugate directions that results in the zero vector:

$$c_0 d_0 + c_1 d_1 + \dots + c_i d_i + \dots + c_k d_k = 0.$$

If we multiply both sides of this equation by $d_i^T Q$ from the left, we obtain

$$c_0 d_i^T Q d_0 + c_1 d_i^T Q d_1 + \dots + c_i d_i^T Q d_i + \dots + c_k d_i^T Q d_k = 0.$$

But the directions d_0, d_1, \dots, d_k are Q -conjugate, so we have

$$c_i d_i^T Q d_i = 0.$$

This means that $c_i = 0$, since Q is positive definite and $d_i \neq 0$. The linear independence of the conjugate directions implies that one can choose at most n Q -conjugate directions.

14/18

Conjugate direction methods for convex quadratic problems

We consider a convex quadratic problem

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

where

$$f(x) = \frac{1}{2} x^T Q x + c^T x,$$

Q is a positive definite matrix. Consider an iteration

$$x_{k+1} = x_k + \alpha_k d_k,$$

where

$$\alpha_k = \arg \min_{\alpha \in \mathbb{R}} f(x_k + \alpha d_k).$$

In the conjugate direction methods the directions d_k are chosen so that

$$\{d_0, d_1, \dots, d_{n-1}\}$$

is a set of Q -conjugate directions.

15/18

Conjugate direction methods for convex quadratic problems

Denote by

$$\phi_k(\alpha) = f(x_k + \alpha d_k).$$

Then

$$\begin{aligned} \phi_k(\alpha) &= \frac{1}{2} (x_k + \alpha d_k)^T Q (x_k + \alpha d_k) + c^T (x_k + \alpha d_k) \\ &= \alpha^2 \left(\frac{1}{2} d_k^T Q d_k \right) + \alpha (x_k^T Q + c^T) d_k + f(x_k) \\ &= \left(\frac{1}{2} d_k^T Q d_k \right) \alpha^2 + (\nabla_k^T d_k) \alpha + f(x_k). \end{aligned}$$

Solving $\phi_k'(\alpha) = 0$, we find that

$$\alpha_k = - \frac{\nabla_k^T d_k}{d_k^T Q d_k},$$

so an iteration of the conjugate direction method in this case is

$$x_{k+1} = x_k - \frac{\nabla_k^T d_k}{d_k^T Q d_k} d_k.$$

16/18

Properties of the conjugate direction methods

1. $\nabla_{k+1}^T d_i = 0, i = 0, \dots, k.$

2. Denote by

$$\alpha^{(k)} = [\alpha_0, \alpha_1, \dots, \alpha_k]^T$$

and by

$$\Phi_k(a^{(k)}) = f(x_0 + a_0 d_0 + a_1 d_1 + \dots + a_k d_k) = f(x_0 + D_k a^{(k)}),$$

where

$$D_k = [d_0 \ d_1 \ \dots \ d_k]$$

is an $n \times (k + 1)$ matrix and

$$a^{(k)} = [a_0, a_1, \dots, a_k]^T$$

is a vector of length $k + 1$. Then we have

$$\alpha^{(k)} = \arg \min_{a^{(k)} \in \mathbb{R}^{k+1}} \Phi(a^{(k)}).$$

17/18

Properties of the conjugate direction methods

Theorem

The conjugate direction algorithm for a convex quadratic function converges to the global minimizer in no more than n steps for any starting point x_0 .

18/18