Second Order Optimization Algorithms I

Yinyu Ye

Department of Management Science and Engineering

Stanford University

Stanford, CA 94305, U.S.A.

Winter 2015

http://www.stanford.edu/~yyye

Chapters 7, 8, 9 and 10

The 1.5-Order Algorithm: Conjugate Gradient Method I

The second-order information is used but no need to inverse it.

0) Initialization: Given initial solution $\mathbf{x}^0$. Let $\mathbf{g}^0 = \nabla f(\mathbf{x}^0)$, $\mathbf{d}^0 = -\mathbf{g}^0$ and $k = 0$.

1) Iterate Update:

$$\mathbf{x}^{k+1} = \mathbf{x}^k + \alpha^k \mathbf{d}^k, \text{ where } \alpha^k = \frac{-(\mathbf{g}^k)^T \mathbf{d}^k}{(\mathbf{d}^k)^T \nabla^2 f(\mathbf{x}^k) \mathbf{d}^k}.$$

2) Compute Conjugate Direction: Compute $\mathbf{g}^{k+1} = \nabla f(\mathbf{x}^{k+1})$. Unless $k = n-1$:

$$\mathbf{d}^{k+1} = -\mathbf{g}^{k+1} + \beta^k \mathbf{d}^k \quad \text{where} \quad \beta^k = \frac{(\mathbf{g}^{k+1})^T \nabla^2 f(\mathbf{x}^k) \mathbf{d}^k}{(\mathbf{d}^k)^T \nabla^2 f(\mathbf{x}^k) \mathbf{d}^k}$$

and set $k = k+1$ and go to Step 1.

3) Restart: Replace $\mathbf{x}^0$ by $\mathbf{x}^n$ and go to Step 0.

For convex quadratic minimization, this process end in no more than 1 round.

The 1.5 Order Algorithm: Conjugate Gradient Method II

The information of the Hessian is learned (more on this later):

0) Initialization: Given initial solution $\mathbf{x}^0$. Let $\mathbf{g}^0 = \nabla f(\mathbf{x}^0)$, $\mathbf{d}^0 = -\mathbf{g}^0$ and $k = 0$.

1) Iterate Update:

$$\mathbf{x}^{k+1} = \mathbf{x}^k + \alpha^k \mathbf{d}^k$$

   where one-dimensional search of α^k is applied.

2) Compute Conjugate Direction: Compute $\mathbf{g}^{k+1} = \nabla f(\mathbf{x}^{k+1})$. Unless $k = n-1$:

$$\mathbf{d}^{k+1} = -\mathbf{g}^{k+1} + \beta^k \mathbf{d}^k$$

$$\text{where } \beta^k = \frac{\|\mathbf{g}^{k+1}\|^2}{\|\mathbf{g}^k\|^2} \text{ or } \beta^k = \frac{(\mathbf{g}^{k+1} - \mathbf{g}^k)^T \mathbf{g}^{k+1}}{\|\mathbf{g}^k\|^2}.$$

   and set $k = k+1$ and go to Step 1.

3) Restart: Replace $\mathbf{x}^0$ by $\mathbf{x}^n$ and go to Step 0.

Bisection Method: First Order Method

For a one variable problem, an KKT point is the root of $g(x) := f'(x) = 0$.

Assume we know an interval $[a\ b]$ such that $a < b$, and $g(a)g(b) < 0$. Then we know there exists an x^*, $a < x^* < b$, such that $g(x^*) = 0$; that is, interval $[a\ b]$ contains a root of g. How do we find x within an error tolerance ϵ, that is, $|x - x^*| \leq \epsilon$?

0) Initialization: let $x_l = a,\ x_r = b$.

1) Let $x_m = (x_l + x_r)/2$, and evaluate $g(x_m)$.

2) If $g(x_m) = 0$ or $x_r - x_l < \epsilon$ stop and output $x^* = x_m$. Otherwise, if $g(x_l)g(x_m) > 0$ set $x_l = x_m$; else set $x_r = x_m$; and return to Step 1.

The length of the new interval containing a root after one bisection step is $1/2$ which gives the linear convergence rate is $1/2$.

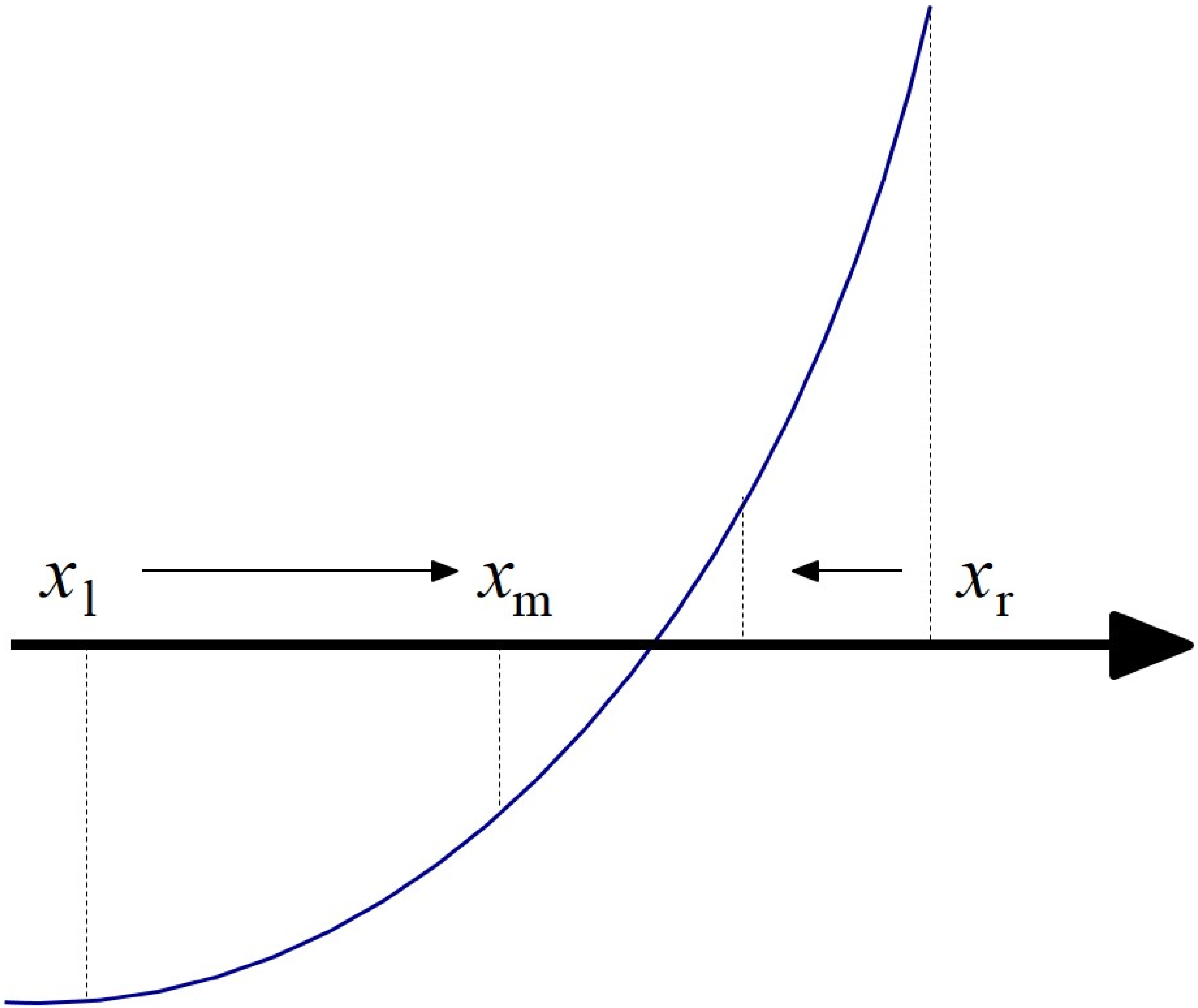

Figure 1: Illustration of Bisection

Golden Section Method: Zero Order Method

Assume that the one variable function $f(x)$ is Unimodel in interval $[a\ b]$, that is, for any point $x \in [a_r\ b_l]$ such that $a \le a_r < b_l \le b$, we have that $f(x) \le \max\{f(a_r),\ f(b_l)\}$. How do we find x^* within an error tolerance ϵ?

0) Initialization: let $x_l = a,\ x_r = b$, and choose a constant $0 < r < 0.5$;

1) Let two other points $\hat{x}_l = x_l + r(x_r - x_l)$ and $\hat{x}_r = x_l + (1 - r)(x_r - x_l)$, and evaluate their function values.

2) Update the triple points $x_r = \hat{x}_r, \hat{x}_r = \hat{x}_l, x_l = x_l$ if $f(\hat{x}_l) < f(\hat{x}_r)$; otherwise update the triple points $x_l = \hat{x}_l, \hat{x}_l = \hat{x}_r, x_r = x_r$; and return to Step 1.

In either cases, the length of the new interval after one golden section step is $(1 - r)$. If we set $(1 - 2r)/(1 - r) = r$, then only one point is new in each step and needs to be evaluated. This give $r = 0.382$ and the linear convergence rate is 0.618.

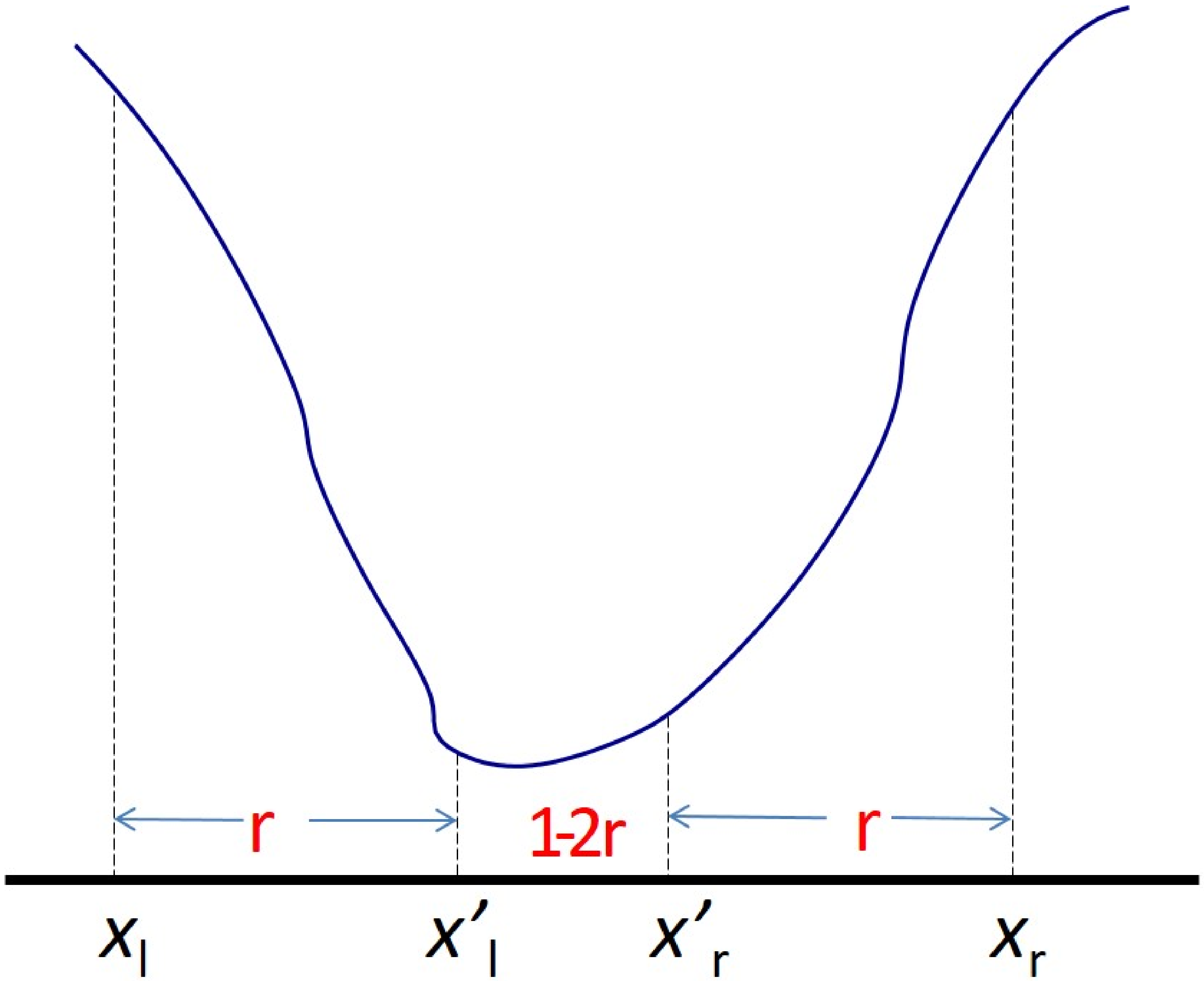

Figure 2: Illustration of Golden Section

Newton's Method: A Second Order Method

For functions of a single real variable x, the KKT condition is $g(x) := f'(x) = 0$. When f is twice continuously differentiable then g is once continuously differentiable, Newton's method can be a very effective way to solve such equations and hence to locate a root of g. Given a starting point x^0, Newton's method for solving the equation $g(x) = 0$ is to generate the sequence of iterates

$$x^{k+1} = x^k - \frac{g(x^k)}{g'(x^k)}.$$

The iteration is well defined provided that $g'(x^k) \neq 0$ at each step.

For multi-variables, Newton's method for minimizing $f(\mathbf{x})$ is defined as

$$\mathbf{x}^{k+1} = \mathbf{x}^k - (\nabla^2 f(\mathbf{x}^k))^{-1} \nabla f(\mathbf{x}^k).$$

We now introduce the second-order β-Lipschitz condition: for any point $\mathbf{x}$ and direction vector $\mathbf{d}$

$$\|\nabla f(\mathbf{x} + \mathbf{d}) - \nabla f(\mathbf{x}) - \nabla^2 f(\mathbf{x})\mathbf{d}\| \leq \beta \|\mathbf{d}\|^2.$$

In the following, for notation simplicity, we use $\mathbf{g}(\mathbf{x}) = \nabla f(\mathbf{x})$ and $\nabla \mathbf{g}(\mathbf{x}) = \nabla^2 f(\mathbf{x})$.

Local Convergence Theorem of Newton's Method

Theorem 1 *Let $f(\mathbf{x})$ be β-Lipschitz and the smallest absolute eigenvalue of its Hessian uniformly bounded below by $\lambda_{min} > 0$. Then, provided that $\|\mathbf{x}^0 - \mathbf{x}^*\|$ is sufficiently small, the sequence generated by Newton's method converges quadratically to $\mathbf{x}^*$ that is a KKT solution with $\mathbf{g}(\mathbf{x}^*) = \mathbf{0}$.*

$$\begin{aligned} \|\mathbf{x}^{k+1} - \mathbf{x}^*\| &= \|\mathbf{x}^k - \mathbf{x}^* - \nabla\mathbf{g}(\mathbf{x}^k)^{-1}\mathbf{g}(\mathbf{x}^k)\| \\ &= \|\nabla\mathbf{g}(\mathbf{x}^k)^{-1}\left(\mathbf{g}(\mathbf{x}^k) - \nabla\mathbf{g}(\mathbf{x}^k)(\mathbf{x}^k - \mathbf{x}^*)\right)\| \\ &= \|\nabla\mathbf{g}(\mathbf{x}^k)^{-1}\left(\mathbf{g}(\mathbf{x}^k) - \mathbf{g}(\mathbf{x}^*) - \nabla\mathbf{g}(\mathbf{x}^k)(\mathbf{x}^k - \mathbf{x}^*)\right)\| \\ &\le \|\nabla\mathbf{g}(\mathbf{x}^k)^{-1}\|\|\mathbf{g}(\mathbf{x}^k) - \mathbf{g}(\mathbf{x}^*) - \nabla\mathbf{g}(\mathbf{x}^k)(\mathbf{x}^k - \mathbf{x}^*)\| \\ &\le \|\nabla\mathbf{g}(\mathbf{x}^k)^{-1}\|\beta\|\mathbf{x}^k - \mathbf{x}^*\|^2 \le \frac{\beta}{\lambda_{min}}\|\mathbf{x}^k - \mathbf{x}^*\|^2. \end{aligned} \tag{1}$$

Thus, when $\frac{\beta}{\lambda_{min}}\|\mathbf{x}^0 - \mathbf{x}^*\| < 1$, the quadratic convergence takes place:

$$\frac{\beta}{\lambda_{min}}\|\mathbf{x}^{k+1} - \mathbf{x}^*\| \le \left(\frac{\beta}{\lambda_{min}}\|\mathbf{x}^k - \mathbf{x}^*\|\right)^2.$$

Such a starting solution $\mathbf{x}^0$ is called an approximate root of $\mathbf{g}(\mathbf{x})$.

How to Check a Point being an Approximate Root

Theorem 2 *(Smale 86). Let $g(x)$ be an analytic function. Then, if x in the domain of g satisfies*

$$\sup_{k>1} \left| \frac{g^{(k)}(x)}{k!g'(x)} \right|^{1/(k-1)} \leq (1/8) \left| \frac{g'(x)}{g(x)} \right| .$$

Then, x is an approximate root of g.

In the following, for simplicity, let the root be in interval $[0 \;\; R]$.

Corollary 1 *(Y. 92). Let $g(x)$ be an analytic function in R^{++} and let g be convex and monotonically decreasing. Furthermore, for $x \in R^{++}$ and $k > 1$ let*

$$\left| \frac{g^{(k)}(x)}{k!g'(x)} \right|^{1/(k-1)} \leq \frac{\alpha}{8} \mathbf{x}^{-1}$$

for some constant $\alpha > 0$. Then, if the root $\bar{x} \in [\hat{x}, (1 + 1/\alpha)\hat{x}] \subset R^{++}$, $\hat{x}$ is an approximate root of g.

Hybrid of Bisection and Newton I

Note that the interval becomes wider and wider at geometric rate when $\hat{x}$ is increased.

Thus, we may symbolically construct a sequence of points:

$$\hat{x}_0 = \epsilon,\ \hat{x}_1 = (1+1/\alpha)\hat{x}_0, ..., \text{ and } \hat{x}_j = (1+1/\alpha)\hat{x}_{j-1}, ...$$

until $\hat{x}_j = \hat{x}_J \geq R$. Obviously the total number of points, J, of these points is bounded by $O(\log(R/\epsilon))$. Moreover, define a sequence of intervals

$$I_j = [\hat{x}_{j-1}, \hat{x}_j] = [\hat{x}_{j-1}, (1+1/\alpha)\hat{x}_{j-1}].$$

Then, if the root $\bar{x}$ of g is in any one of these intervals, say in I_j, then the front point $\hat{x}_{j-1}$ of the interval is an approximate root of g so that starting from it Newton's method generates an x with $|x - \bar{x}| \leq \epsilon$ in $O(\log\log(1/\epsilon))$ iterations.

Hybrid of Bisection and Newton II

Now the question is how to identify the interval that contains $\bar{x}$?

This time, we bisect the number of intervals, that is, evaluate function value at point $\hat{x}_{j_m}$ where $j_m = [J/2]$. Thus, each bisection reduces the total number of the intervals by a half. Since the total number of intervals is $O(\log(R/\epsilon))$, in at most $O(\log\log(R/\epsilon))$ bisection steps we shall locate the interval that contains $\bar{x}$.

Then the total number iterations, including both bisection and Newton methods, is $O(\log\log(R/\epsilon))$ iterations.

Here we take advantage of the global convergence property of Bisection and local quadratic convergence property of Newton, and we would see more of these features later...

Spherical Constrained Nonconvex Quadratic Minimization I

$$\min \ \frac{1}{2}\mathbf{x}^T Q\mathbf{x} + \mathbf{c}^T\mathbf{x}, \quad \text{s.t.} \quad \|\mathbf{x}\|^2 = 1.$$

where $Q \in S^n$ is any symmetric data matrix. If $\mathbf{c} = \mathbf{0}$ this problem becomes finding the least eigenvalue of Q.

The necessary and sufficient condition (can be proved using SDP) for $\mathbf{x}$ being a global minimizer of the problem is

$$(Q + \lambda I)\mathbf{x} = -\mathbf{c}, \ (Q + \lambda I) \succeq \mathbf{0}, \ \|\mathbf{x}\|_2^2 = 1,$$

which implies $\lambda \geq -\lambda_{min}(Q) > 0$ where $\lambda_{min}(Q)$ is the least eigenvalue of Q. If the optimal $\lambda^* = -\lambda_{min}(Q)$, then $\mathbf{c}$ must be orthogonal to the $\lambda_{min}(Q)$-eigenvector, and it can be checked using the power algorithm.

The minimal objective value:

$$\frac{1}{2}\mathbf{x}^T Q\mathbf{x} + \mathbf{c}^T\mathbf{x} = -\frac{1}{2}\mathbf{x}^T(Q + \lambda I)\mathbf{x} - \frac{1}{2}\lambda\|\mathbf{x}\|^2 = -\frac{\lambda}{2}, \tag{2}$$

Sphere Constrained Nonconvex Quadratic Minimization II

WLOG, Let us assume that the least eigenvalue is 0. Then we must have $\lambda \geq 0$. If the optimal $\lambda^* = 0$, then $\mathbf{c}$ must be a 0-eigenvector of Q, and it can be checked using the power algorithm to find it. Therefore, we assume that the optimal $\lambda > 0$.

Furthermore, there is an upper bound on λ:

$$\lambda \leq \lambda \|\mathbf{x}\|^2 \leq \mathbf{x}^T(Q + \lambda I)\mathbf{x} = -\mathbf{c}^T\mathbf{x} \leq \|\mathbf{c}\|\|\mathbf{x}\| = \|\mathbf{c}\|.$$

Now let $\mathbf{x}(\lambda) = -(Q + \lambda I)^{-1}\mathbf{c}$, the problem becomes finding the root of $\|\mathbf{x}(\lambda)\|^2 = 1$.

Lemma 1 *The analytic function $\|\mathbf{x}(\lambda)\|^2$ is convex monotonically decreasing with $\alpha = 12$ in Corollary 1.*

Theorem 3 *The 1-spherical constrained quadratic minimization can be computed in $O(\log\log(\|\mathbf{c}\|/\epsilon))$ iterations where each iteration costs $O(n^3)$ arithmetic operations.*

What about 2-spherical constrained quadratic minimization, that is, quadratic minimization with 2 ellipsoidal constraints?

Second Order Method for Minimizing Lipschitz $f(\mathbf{x})$

Recall the second-order β-Lipschitz condition: for any two points $\mathbf{x}$ and $\mathbf{y}$

$$\|\mathbf{g}(\mathbf{x}+\mathbf{d}) - \mathbf{g}(\mathbf{x}) - \nabla\mathbf{g}(\mathbf{x})\mathbf{d}\| \le \beta\|\mathbf{d}\|^2,$$

which further implies

$$f(\mathbf{x}+\mathbf{d}) - f(\mathbf{x}) \le \mathbf{g}(\mathbf{x})^T\mathbf{d} + \frac{1}{2}\mathbf{d}^T\nabla\mathbf{g}(\mathbf{x})\mathbf{d} + \frac{\beta}{3}\|\mathbf{d}\|^3.$$

The second-order method, at the kth iterate, would let $\mathbf{x}^{k+1} = \mathbf{x}^k + \mathbf{d}^k$ where

$$\begin{array}{rll} \mathbf{d}^k = & \arg\min_{\mathbf{d}} & (\mathbf{c}^k)^T\mathbf{d} + \frac{1}{2}\mathbf{d}^TQ^k\mathbf{d} + \frac{\beta}{3}\alpha^3 \\ & \text{s.t.} & \|\mathbf{d}\| \le \alpha, \end{array}$$

with $\mathbf{c}^k = \mathbf{g}(\mathbf{x}^k)$ and $Q^k = \nabla\mathbf{g}(\mathbf{x}^k)$. One typically fixed α to a "trusted' radius α^k so that it becomes a sphere-constrained problem (the inequality is normally active if the Hessian is non PSD):

$$(Q^k + \lambda^k I)\mathbf{d}^k = -\mathbf{c}^k,\ (Q^k + \lambda^k I) \succeq \mathbf{0},\ \|\mathbf{d}^k\|_2^2 = (\alpha^k)^2.$$

Convergence Speed of the Second Order Method

A naive choice would be $\alpha^k = \sqrt{\epsilon}/\beta$. Then from reduction (2)

$$f(\mathbf{x}^{k+1}) - f(\mathbf{x}^k) \le -\frac{\lambda^k}{2}\|\mathbf{d}^k\|^2 + \frac{\beta}{3}(\alpha^k)^3 = -\frac{\lambda^k(\alpha^k)^2}{2} + \frac{\beta}{3}(\alpha^k)^3 = -\frac{\lambda^k\epsilon}{2\beta^2} + \frac{\epsilon^{3/2}}{3\beta^2}.$$

Also

$$\begin{aligned}\|\mathbf{g}(\mathbf{x}^{k+1})\| &= \|\mathbf{g}(\mathbf{x}^{k+1}) - (\mathbf{c}^k + Q^k\mathbf{d}^k) + (\mathbf{c}^k + Q^k\mathbf{d}^k)\| \\ &\le \|\mathbf{g}(\mathbf{x}^{k+1}) - (\mathbf{c}^k + Q^k\mathbf{d}^k)\| + \|(\mathbf{c}^k + Q^k\mathbf{d}^k)\| \\ &\le \beta\|\mathbf{d}^k\|^2 + \lambda^k\|\mathbf{d}^k\| = \beta(\alpha^k)^2 + \lambda^k\alpha^k = \frac{\epsilon}{\beta} + \frac{\lambda^k\sqrt{\epsilon}}{\beta}.\end{aligned}$$

Thus, one can stop the algorithm as soon as $\lambda^k = \sqrt{\epsilon}$ so that the inequality becomes $\|\mathbf{g}(\mathbf{x}^{k+1})\| \le \frac{2\epsilon}{\beta}$. Furthermore, $|\lambda_{min}(\nabla\mathbf{g}(\mathbf{x}^k))| \le \lambda^k = \sqrt{\epsilon}$.

Theorem 4 *Let the objective function $p^* = \inf\ f(\mathbf{x})$ be finite. Then in $\frac{O(\beta^2(f(\mathbf{x}^0)-p^*))}{\epsilon^{1.5}}$ iterations of the second-order method, the norm of the gradient vector is less than ϵ and the Hessian is $\sqrt{\epsilon}$-positive semidefinite.*

Would Convexity Help?

Before we answer this question, let's summarize a generic form one iteration of the Second Order Method for solving $\nabla f(\mathbf{x}) = \mathbf{g}(\mathbf{x}) = \mathbf{0}$:

$$(\nabla \mathbf{g}(\mathbf{x}^k) + \lambda I)(\mathbf{x} - \mathbf{x}^k) = -\gamma \mathbf{g}(\mathbf{x}^k), \quad \text{or}$$
$$\mathbf{g}(\mathbf{x}^k) + \nabla \mathbf{g}(\mathbf{x}^k)(\mathbf{x} - \mathbf{x}^k) + \lambda(\mathbf{x} - \mathbf{x}^k) = (1-\gamma)\mathbf{g}(\mathbf{x}^k).$$

Many interpretations: when

- $\gamma = 1, \lambda = 0$: pure Newton;
- γ and λ are sufficiently large: SDM;
- $\gamma = 1$ and λ decreases to 0: Homotopy or path-following method.

The Quasi-Newton Method More generally:

$$\mathbf{x} = \mathbf{x}^k - \alpha^k S^k \mathbf{g}(\mathbf{x}^k),$$

for a symmetric matrix S^k with a step-size α^k.

The Quasi-Newton Method

For convex qudratic minimization, the convergnece rate becomes $\left(\frac{\lambda_{max}(S^kQ)-\lambda_{min}(S^kQ)}{\lambda_{max}(S^kQ)+\lambda_{min}(S^kQ)}\right)^2$ where λ_{max} and λ_{min} represent the largest and smallest eigenvalues of a matrix.

S^k can be viewed as a Preconditioner–typically an approximation of the Hessian matrix inverse, and can be learned from a regression model:

$$\mathbf{q}^k := \mathbf{g}(\mathbf{x}^{k+1}) - \mathbf{g}(\mathbf{x}^k) = Q(\mathbf{x}^{k+1} - \mathbf{x}^k) = Q\mathbf{d}^k,\ k = 0, 1, ...$$

We actually learn Q^{-1} from $Q^{-1}\mathbf{q}^k = \mathbf{d}_k,\ k = 0, 1, ...$ The process start with H^k, $k = 0, 1, ...$, where the rank of H^k is k, that is, we each step lean a rank-one update: given $H^{k-1},\ \mathbf{q}^k,\ \mathbf{d}^k$ we solve

$$(H^{k-1} + \mathbf{h}^k(\mathbf{h}^k)^T)\mathbf{q}^k = \mathbf{d}^k$$

for vector $\mathbf{h}^k$. Then after n iterations, we build up $H^n = Q^{-1}$.

You also "learnig while doing": $\mathbf{x}^{k+1} = \mathbf{x}^k - \alpha^k(\frac{n-k}{n}I + \frac{k}{n}H^k)\mathbf{g}(\mathbf{x}^k)$, which is similar to the Conjugate Gradient method.

We now give a confirmation answer: convexity helps a lot in Second-Order methods.

A Path-Following Algorithm for Unconstrained Optimization I

We assume that f is convex and meet a local Lipschitz condition: for any point $\mathbf{x}$ and a $\beta \geq 1$

$$\|\mathbf{g}(\mathbf{x}+\mathbf{d}) - \mathbf{g}(\mathbf{x}) - \nabla \mathbf{g}(\mathbf{x})\mathbf{d}\| \leq \beta \mathbf{d}^T \nabla \mathbf{g}(\mathbf{x})\mathbf{d}, \text{ whenever } \|\mathbf{d}\| \leq O(1) \tag{3}$$

and $\mathbf{x}+\mathbf{d}$ in the function domain. We start from a solution $\mathbf{x}^k$ that approximately satisfies

$$\mathbf{g}(\mathbf{x}) + \lambda \mathbf{x} = \mathbf{0}, \quad \text{with} \quad \lambda = \lambda^k > 0. \tag{4}$$

Such a solution $\mathbf{x}(\lambda)$ exists for any $\lambda > 0$ because it is the (unique) optimal solution for problem

$$\mathbf{x}(\lambda) = \arg\min \ f(\mathbf{x}) + \frac{\lambda}{2}\|\mathbf{x}\|^2,$$

and they form a path down to $\mathbf{x}(0)$. Let the approximation path error at $\mathbf{x}^k$ with $\lambda = \lambda^k$ be

$$\|\mathbf{g}(\mathbf{x}^k) + \lambda^k \mathbf{x}^k\| \leq \frac{1}{2\beta}\lambda^k.$$

Then, we like to compute a new iterate $\mathbf{x}^{k+1}$ such that

$$\|\mathbf{g}(\mathbf{x}^{k+1}) + \lambda^{k+1}\mathbf{x}^{k+1}\| \leq \frac{1}{2\beta}\lambda^{k+1}, \quad \text{where } 0 \leq \lambda^{k+1} < \lambda^k.$$

A Path-Following Algorithm for Unconstrained Optimization II

When λ^k is replaced by λ^{k+1}, say $(1-\eta)\lambda^k$ for some $\eta \in (0,\ 1]$, we aim to find a solution $\mathbf{x}$ such that

$$\mathbf{g}(\mathbf{x}) + (1-\eta)\lambda^k \mathbf{x} = \mathbf{0},$$

we start from $\mathbf{x}^k$ and apply the Newton iteration:

$$\begin{array}{c} \mathbf{g}(\mathbf{x}^k) + \nabla \mathbf{g}(\mathbf{x}^k)\mathbf{d} + (1-\eta)\lambda^k(\mathbf{x}^k + \mathbf{d}) = \mathbf{0}, \quad \text{or} \\ \nabla \mathbf{g}(\mathbf{x}^k)\mathbf{d} + (1-\eta)\lambda^k \mathbf{d} = -\mathbf{g}(\mathbf{x}^k) - (1-\eta)\lambda^k \mathbf{x}^k. \end{array} \tag{5}$$

From the second expression, we have

$$\begin{array}{rcl} \|\nabla \mathbf{g}(\mathbf{x}^k)\mathbf{d} + (1-\eta)\lambda^k \mathbf{d}\| & = & \|-\mathbf{g}(\mathbf{x}^k) - (1-\eta)\lambda^k \mathbf{x}^k\| \\ & = & \|-\mathbf{g}(\mathbf{x}^k) - \lambda^k \mathbf{x}^k + \eta\lambda^k \mathbf{x}^k\| \\ & \le & \|-\mathbf{g}(\mathbf{x}^k) - \lambda^k \mathbf{x}^k\| + \eta\lambda^k\|\mathbf{x}^k\| \\ & \le & \frac{1}{2\beta}\lambda^k + \eta\lambda^k\|\mathbf{x}^k\|. \end{array} \tag{6}$$

On the other hand

$$\|\nabla \mathbf{g}(\mathbf{x}^k)\mathbf{d} + (1-\eta)\lambda^k \mathbf{d}\|^2 = \|\nabla \mathbf{g}(\mathbf{x}^k)\mathbf{d}\|^2 + 2(1-\eta)\lambda^k \mathbf{d}^T \nabla \mathbf{g}(\mathbf{x}^k)\mathbf{d} + ((1-\eta)\lambda^k)^2\|\mathbf{d}\|^2.$$

From convexity, $\mathbf{d}^T \|\nabla \mathbf{g}(\mathbf{x}^k)\mathbf{d} \geq 0$, together with (6) we have

$$\begin{aligned} ((1-\eta)\lambda^k)^2\|\mathbf{d}\|^2 &\leq (\tfrac{1}{2\beta} + \eta\|\mathbf{x}^k\|)^2(\lambda^k)^2 \quad \text{and} \\ 2(1-\eta)\lambda^k \mathbf{d}^T \|\nabla \mathbf{g}(\mathbf{x}^k)\mathbf{d} &\leq (\tfrac{1}{2\beta} + \eta\|\mathbf{x}^k\|)^2(\lambda^k)^2. \end{aligned}$$

The first inequality implies

$$\|\mathbf{d}\|^2 \leq (\frac{1}{2\beta(1-\eta)} + \frac{\eta}{1-\eta}\|\mathbf{x}^k\|)^2.$$

Let the new iterate be $\mathbf{x}^+ = \mathbf{x}^k + \mathbf{d}$. The second inequality implies

$$\begin{aligned} & \|\mathbf{g}(\mathbf{x}^+) + (1-\eta)\lambda^k \mathbf{x}^+\| \\ = \; & \|\mathbf{g}(\mathbf{x}^+) - (\mathbf{g}(\mathbf{x}^k) + \nabla \mathbf{g}(\mathbf{x}^k)\mathbf{d}) + (\mathbf{g}(\mathbf{x}^k) + \nabla \mathbf{g}(\mathbf{x}^k)\mathbf{d}) + (1-\eta)\lambda^k(\mathbf{x}^k + \mathbf{d})\| \\ = \; & \|\mathbf{g}(\mathbf{x}^+) - \mathbf{g}(\mathbf{x}^k) + \nabla \mathbf{g}(\mathbf{x}^k)\mathbf{d}\| \\ \leq \; & \beta \mathbf{d}^T \nabla \mathbf{g}(\mathbf{x}^k)\mathbf{d} \leq \tfrac{\beta}{2(1-\eta)}(\tfrac{1}{2\beta} + \eta\|\mathbf{x}^k\|)^2\lambda^k. \end{aligned}$$

We now just need to choose $\eta \in (0,\ 1)$ such that

$$\begin{aligned} \left(\frac{1}{2\beta(1-\eta)} + \frac{\eta}{1-\eta}\|\mathbf{x}^k\|\right)^2 &\le 1 \quad \text{and} \\ \frac{\beta\lambda^k}{2(1-\eta)}\left(\frac{1}{2\beta} + \eta\|\mathbf{x}^k\|\right)^2 &\le \frac{1}{2\beta}(1-\eta)\lambda^k = \frac{1}{2\beta}\lambda^{k+1}. \end{aligned}$$

For example, given $\beta \ge 1$,

$$\eta = \frac{1}{2\beta(1+\|\mathbf{x}^k\|)}$$

would suffice.

This would give a linear convergence since $\|\mathbf{x}^k\|$ is typically bounded following the path to the optimality, while the covergence in non-convex case is only arithmetic.

Convexity, together with some types of second-order methods, make convex optimization solvers into practical technoloies.