Optimization Methods
Lecture 8

Solmaz S. Kia
Mechanical and Aerospace Engineering Dept.
University of California Irvine
solmaz@uci.edu

Reading: page 285-297 from Ref[2].

Numerical solvers for unconstrained optimization

Unconstrained optimization:

$$x^\star = \underset{x\in\mathbb{R}^n}{\text{argmin}}\ f(x)$$

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

Observations:

- Steepest descent algorithm can be very slow with lots of zig-zaging
- Newton method is faster but numerically is expensive due to information equipment associated with the evaluation, storage and inversion of Hessian.

Q: Is it possible to accelerate convergence with low numerical cost?

A: Quasi-Newton methods: Consider $x_{k+1} = x_k - \alpha_k\, S_k\, g_k$

- Try to construct the inverse Hessian, or an approximation of it, using information gathered as the descent process progresses.

- The current approximation H_k is then used at each stage to define the next descent direction by setting $S_k = H_k$ in the modified Newton method.

Quasi Newton Methods (review from last week)

Let

- $g_k = \nabla f(x_k)$,
- $q_k = g_{k+1} - g_k$,
- $p_k = x_{k+1} - x_k$

then $g(x_{k+1}) = g(x_k + p_k) \approx g(x_k) + \nabla^2 f(x_k)^\top p_k$. Therefore,

$$q_k \approx \nabla^2 f(x_k)\, p_k$$

or

$$(\nabla^2 f(x_k))^{-1} q_k \approx p_k$$

We expect that H_k that wants to approximate $(\nabla^2 f(x_k))^{-1}$ should satisfy

1. $H_{k+1} q_i = p_i, \quad i \in \{0, 1, \cdots, k\}$
2. H_k symmetric
3. $H_k > 0$

For the case of constant Hessian, after n linearly independent steps, then we have $H_n = F^{-1}$.

Quasi Newton Methods (review from last week)

Initialization $k = 0$: start by $x_0 \in^n$ and any $H_0 > 0$
Step 1. Set $d_k = -H_k g_k$.
Step 2. obtain $\alpha_k = \subset \alpha > 0 \text{argmin} f(x_k + \alpha d_k)$. Then obtain $x_{k+1} = x_k + \alpha d_k$ and $p_k = \alpha_k d_k$, and g_{k+1}.
Step 3. Set $q_k = g_{k+1} - g_k$ and

$$\text{Rank one correction:} H_{k+1} = H_k + \frac{(p_k - H_k\, q_k)(p_k - H_k\, q_k)^\top}{q_k^\top (p_k - H_k q_k)}$$

$$\text{DFP method :} H_{k+1} = H_k + \frac{p_k p_k^\top}{p_k^\top q_k} - \frac{H_k q_k q_k^\top H_k}{q_k^\top H_k q_k}.$$

Check the stoping condition; if not satisfied update k and return to Step 1.

In **Rank One Correction**

- H_k is symmetric
- But not necessarily positive definite (we need $q_k^\top (p_k - H_k q_k) > 0$ which is not guaranteed at all times).

DFP method generates positive definite H_k and has better convergence results that the Rank One Correction method.

Quasi Newton Methods: The Broyden family

The idea in the Broyden method is to first approximate the Hessian (denote this estimate by B_k) and then inverse it to obtain the inverse Hessian approximation (denote this estimate by H_k) which will be use in the quasi-Newton method to compute the $x_{k+1} = x_k - \alpha_k H_k g(x_k)$, where $H_k = (B_k)^{-1}$. Recall

- $g_k = \nabla f(x_k)$, $q_k = g_{k+1} - g_k$ and $p_k = x_{k+1} - x_k$

then $g(x_{k+1}) = g(x_k + p_k) \approx g(x_k) + \nabla^2 f(x_k)^\top p_k$. Therefore, $q_k \approx \nabla^2 f(x_k)\, p_k$. We expect that B_k that wants to approximate $(\nabla^2 f(x_k))$ should satisfy

1. $B_{k+1} p_i = q_i, \quad i \in \{0, 1, \cdots, k\}$
2. B_k symmetric and $B_k > 0$

For constant Hessian F, after n linearly independent steps, then we have $B_n = F$.

To develop the Broyden approximate to the Hessian, we follow the DFP method exactly with the only difference that q_p and p_k are replaced, replaced respectively by p_k and q_k.

$$\text{DFP method}: \; H_{k+1} = H_k + \frac{p_k p_k^\top}{p_k^\top q_k} - \frac{H_k q_k q_k^\top H_k}{q_k^\top H_k q_k}$$

$$\text{Broyden-Fletcher-Godfarb-Shanno (BFGS) method}: \; B_{k+1} = B_k + \frac{q_k q_k^\top}{q_k^\top p_k} - \frac{B_k p_k p_k^\top B_k}{p_k^\top B_k p_k}$$

Starting with a $B_0 > 0$, similar B_k is guaranteed to be positive definite for $k > 0$.

Quasi Newton Methods: The Broyden family

$$B_{k+1} = B_k + \frac{q_k q_k^\top}{q_k^\top p_k} - \frac{B_k p_k p_k^\top B_k}{p_k^\top B_k p_k}$$

We are interested in $H_k = (B_k)^{-1}$. As it happens we can use the property below to compute H_k in a closed form.

Sherman-Morrison formula: Let $A \in \mathbb{R}^{n \times n}$ be invertible. Then, for $a \in \mathbb{R}^n$ and $b \in \mathbb{R}^n$ we have

$$\left(A + a\,b^\top\right)^{-1} = A^{-1} - \frac{A^{-1} a\, b^\top A^{-1}}{1 + b^\top A^{-1} a}.$$

$$H_{k+1}^{BFGS} = (B_{k+1}^{BFGS})^{-1} = H_k + \left(1 + \frac{q_k^\top H_k q_k}{p_k^\top q_k}\right) \frac{p_k p_k^\top}{p_k^\top q_k} - \frac{H_k q_k p_k^\top + p_k q_k^\top H_k}{p_k^\top q_k}$$

- Numerical experiments have repeatedly shown that BFGS has superior performance in comparison to the DFP method.

Quasi Newton Methods: The Broyden family

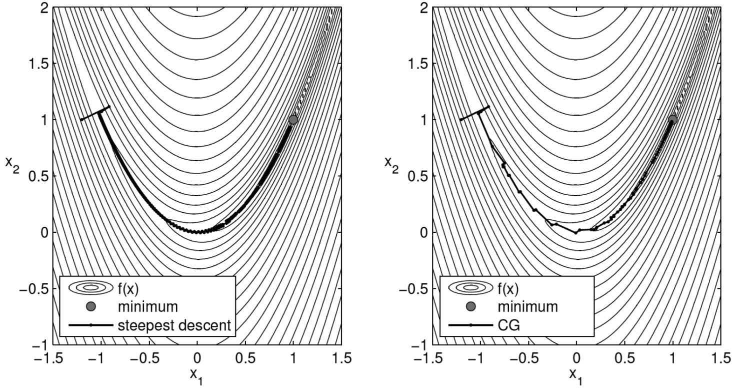
- Broyden family update is obtained from combining the BFGS and the DFP method
$$H^{\phi} = (1-\phi)H^{DFP} + \phi H^{BFGS}$$
where ϕ can take any value.
- An explicit representation of Broyden family can be shown to be
$$H_{k+1}^{\phi} = H_k + \frac{p_k p_k^\top}{p_k^\top q_k} - \frac{H_k q_k q_k^\top H_k}{q_k^\top H_k q_k} + \phi \tau_k v_k v_k^\top = H_{k+1}^{DFP} + \phi v_k v_k^\top$$
where $v_k = \frac{p_k}{p_k^\top q_k} - \frac{H_k q_k}{\tau_k}$ and $\tau_k = q_k^\top H_k q_k$
- The parameter ϕ is, in general, allowed to vary from one iteration to another
- A Broyden family is defined. by a sequence ϕ_1, ϕ_2, $\cdots$, of parameter values.
- A pure Broyden method is one that uses a constant ϕ
- For $\phi = 0$ we recover the DFP method
- For $\phi = 1$ we recover the BFGS method
- For $0 \leqslant \phi \leqslant 1$, H^{ϕ} is positive definite
- For $\phi < 0$ and $\phi > 1$ there is possibility that H^{ϕ} may become singular
- In practice $0 \leqslant \phi \leqslant 1$ is usually imposed to avoid difficulties

Numerical example

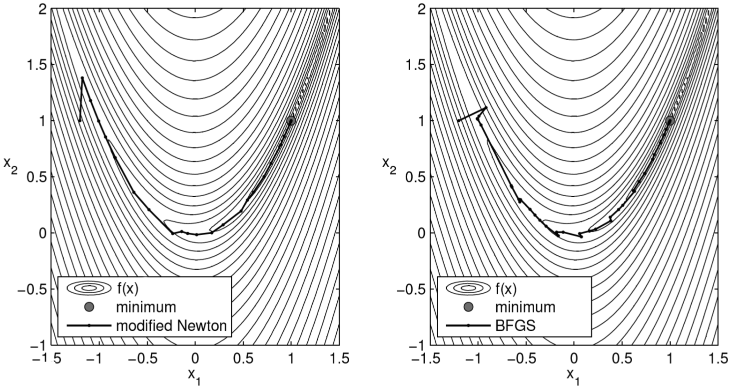
Minimize Rosenbrock's function,

$$f(x) = 100\left(x_2 - x_1^2\right)^2 + (1 - x_1)^2 ,$$

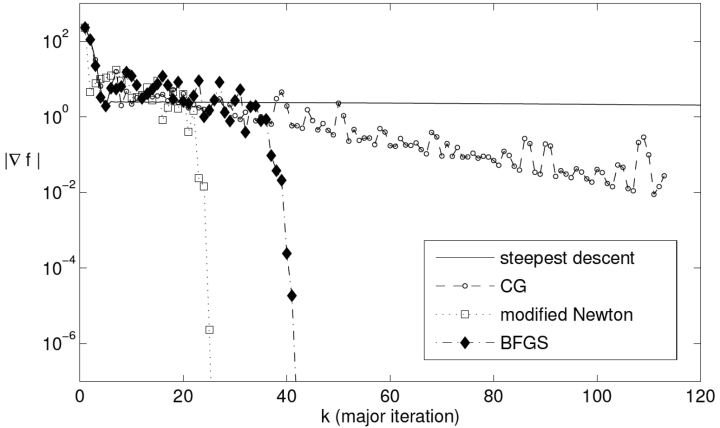
starting from $x_0 = (-1.2, 1.0)^T$.



Solution path of the steepest descent and conjugate gradient methods



Solution path of the modified Newton and BFGS methods



Comparison of convergence rates for the Rosenbrock function

Trust Region (restricted-step)methods

- Trust region, or "restricted-step" methods are a different approach to resolving the weaknesses of the pure form of Newton's method, arising from an Hessian that is not positive definite or a highly nonlinear function.
- One way to interpret these problems is to say that they arise from the fact that we are stepping outside a the region for which the quadratic approximation is reasonable. Thus we can overcome this difficulties by minimizing the quadratic function within a region around x_k within which we trust the quadratic model.

Consider $x_{k+1} = x_k + p_k$. The algorithm in the next slide we design p_k using a Trust Region method. Note that there are different variations of the Trust Region method. Here we only present one of these method.

A Trust Region algorithm

1. Select x_0 and a convergence parameter $\epsilon > 0$ and the initial size of the trust region, h_0.
2. Compute $g(x_k) = \nabla f(x_k)$. If $\|g(x_k)\| \leqslant \epsilon$ then stop. Otherwise, continue.
3. Compute $H(x_k) = \nabla^2 f(x_k)$ and solve the quadratic subproblem

$$p_k = \underset{p \in \mathbb{R}^n}{\operatorname{argmin}}\, q(p) = f(x_k) + g(x_k)^\top p + \frac{1}{2} p^\top H(x_k) p, \quad \text{s.t.}$$

$$-h_k \leqslant p^i \leqslant h_k, i = 1, \cdots, n, \qquad (p^i \text{ is the ith element of } p \in \mathbb{R}^n)$$

4. Compute the ratio that measures the accuracy of the quadratic model,

$$r_k = \frac{\overbrace{f(x_k) - f(x_k + p_k)}^{\text{actual function reduction}}}{\underbrace{q(0) - q(p_k)}_{\text{predicted function reduction}}} = \frac{f(x_k) - f(x_k + p_k)}{f(x_k) - q(p_k)}$$

5. Compute the size for the new trust region as follows:

$$h_{k+1} = \begin{cases} \frac{\|p_k\|}{4} & \text{if } r_k < 0.25, \\ 2h_k & \text{if } r_k > 0.75 \text{ and } h_k = \|p_k\|, \\ h_k, & \text{otherwise.} \end{cases}$$

6. Determine the new point: $x_{k+1} = \begin{cases} x_k & \text{if } r_k \leqslant 0, \\ x_k + p_k & \text{otherwise,} \end{cases}$
7. Set $k = k + 1$ and return to 2.

Note: The initial value of h is usually 1. The same stopping criteria used in other gradient-based methods are also applicable.