Numerical Optimization 08: Quasi-Newton methods

Qiang Zhu

University of Nevada Las Vegas

July 14, 2020

Overview

Quasi-Newton's method

Just as the secant method approximates f" in the univariate case, quasi Newton approximate the inverse Hessian $((\boldsymbol{H}^k)^{-1})$ which is needed for each step of update

$$\boldsymbol{x}^{k+1} \leftarrow \boldsymbol{x}^k - \alpha^k (\boldsymbol{H}^k)^{-1} \boldsymbol{g}^k$$

These methods typically set $(\boldsymbol{H}^k)^{-1}$ (let's call it $\boldsymbol{Q}$ from now on) to the identity matrix and then apply updates to reflect information learned with each iteration. To simplify the equations for various quasi-Newton methods, we define the following

$$\begin{aligned} \boldsymbol{\gamma}^{k+1} &= \boldsymbol{g}^{k+1} - \boldsymbol{g}^k \\ \boldsymbol{\delta}^{k+1} &= \boldsymbol{x}^{k+1} - \boldsymbol{x}^k \end{aligned}$$

A new quadratic model

Instead of computing the exact $\boldsymbol{Q}$, we can update it in a simple manner to account for the curvature measured during the most recent step. Suppose, we have generated $\boldsymbol{x}^{k+1}$ and wish to construct a new quadratic model,

$$m^{k+1}(\boldsymbol{p}) = f(x^{k+1}) + \boldsymbol{g}^{k+1}\boldsymbol{p} + \frac{1}{2}\boldsymbol{p}^T\boldsymbol{Q}^{k+1}\boldsymbol{p}$$

We let the gradient of m^{k+1} match the gradient of f for at least two steps $\boldsymbol{x}^{k+1}$ and $\boldsymbol{x}^k$.

$$\nabla m^{k+1}(-\alpha^k p^k) = \boldsymbol{g}^{k+1} - \alpha^k\boldsymbol{Q}^{k+1}\boldsymbol{p}^k = \boldsymbol{g}^k$$

Since $\nabla m^{k+1}(0) = g^{k+1}$, the second of these condition is satisfied automatically. Rearranging it, we obtain the so called secant condition.

$$\boldsymbol{Q}^{k+1}\alpha^k\boldsymbol{p}^k = \boldsymbol{g}^{k+1} - \boldsymbol{g}^k \quad \rightarrow \quad \boldsymbol{Q}^{k+1}\boldsymbol{\delta}^k = \boldsymbol{\gamma}^k \tag{1}$$

A new quadratic model

Given the displacements $\boldsymbol{\delta}^k$ and the change of gradients γ^k. It requires that the symmetric positive definite matrix $\boldsymbol{Q}^{k+1}$, it needs that

$$\boldsymbol{\delta}^k \gamma^k > 0$$

At this stage, there still exists an infinite number of solutions of $\boldsymbol{Q}^{k+1}$. To determine a unique solution, we impose another condition, which is that $\boldsymbol{Q}^{k+1}$ is close to the current $\boldsymbol{Q}^k$

$$\min_{\boldsymbol{Q}} ||\boldsymbol{Q} - \boldsymbol{Q}^k||$$
$$\text{s.t.} \qquad \boldsymbol{Q} = \boldsymbol{Q}^T, \qquad \boldsymbol{B}\delta^k = \gamma^k$$

Different matrix norms can be applied here to give different quasi-Newton methods.

The Davidon-Fletcher-Powell (DFP) method

Davidon proposed a relation to ensures that $\boldsymbol{Q}$ is symmetric.

$$\boldsymbol{Q}^{k+1} = \boldsymbol{Q}^k + auu^T + bvv^T$$

According to the secant condition

$$\boldsymbol{Q}^k \boldsymbol{\delta}^k + auu^T \boldsymbol{\delta}^k + bvv^T \boldsymbol{\delta}^k = \gamma^k$$

An obvious choice for u and v is

$$u = \gamma^k, \qquad v = \boldsymbol{Q}^k \boldsymbol{\delta}^k \quad \rightarrow \quad au^T \boldsymbol{\delta}^k = 1, \quad bv^T \boldsymbol{\delta}^k = -1$$

where

$$a = 1/u^T \boldsymbol{\delta}^k = 1/u^T \boldsymbol{\delta}^k \quad b = -1/v^T \boldsymbol{\delta}^k = 1/u^T \boldsymbol{\delta}^k$$

$$\boldsymbol{Q}^{k+1} = \boldsymbol{Q}^k - \frac{\boldsymbol{Q}^k \gamma^k (\gamma^k)^T \boldsymbol{Q}^k}{(\gamma^k)^T \boldsymbol{Q}^k \gamma^k} + \frac{\boldsymbol{\delta}(\boldsymbol{\delta}^k)^T}{(\boldsymbol{\delta}^k)^T \gamma^k}$$

W. C. Davidon, Variable Metric Method for Minimization
SIAM Journal on Optimization. 1. (1991), 1-17.

The Broyden-Fletcher-Goldfarb-Shanno (BFGS) method

The BFGS algorithm does not approximate $\boldsymbol{Q}^k$, but handles $\boldsymbol{H}^k = (\boldsymbol{Q}^k)^{-1}$

$$\boldsymbol{H}^{k+1}\gamma^k = \boldsymbol{\delta}^k$$

The minimize condition is,

$$\begin{aligned} &\min_{\boldsymbol{H}} ||\boldsymbol{H} - \boldsymbol{H}^k|| \\ \text{s.t.} \qquad &\boldsymbol{H} = \boldsymbol{H}^T, \qquad \boldsymbol{H}\gamma^k = \boldsymbol{\delta}^k \end{aligned}$$

$$\boldsymbol{Q}^{k+1} = \boldsymbol{Q}^k - \frac{\boldsymbol{\delta}^k(\gamma^k)^T\boldsymbol{Q}^k + \boldsymbol{Q}^k\gamma^k(\boldsymbol{\delta}^{\boldsymbol{k}})^T}{(\boldsymbol{\delta}^k)^T\gamma^k} + \left(1 + \frac{(\gamma^k)^T\boldsymbol{Q}^k\gamma^k}{(\boldsymbol{\delta}^k)^T\boldsymbol{Q}^k}\right)\frac{\boldsymbol{\delta}^k(\boldsymbol{\delta}^k)^T}{(\boldsymbol{\delta}^k)^T\gamma^k}$$

BFGS does better than DFP with approximate line search.

The Broyden-Fletcher-Goldfarb-Shanno (BFGS) method

The BFGS algorithm does not approximate $\boldsymbol{Q}^k$, but handles $\boldsymbol{H}^k = (\boldsymbol{Q}^k)^{-1}$

$$\boldsymbol{H}^{k+1}\gamma^k = \boldsymbol{\delta}^k$$

The minimize condition is,

$$\min_{\boldsymbol{H}} ||\boldsymbol{H} - \boldsymbol{H}^k||$$
$$\text{s.t.} \qquad \boldsymbol{H} = \boldsymbol{H}^T, \qquad \boldsymbol{H}\gamma^k = \boldsymbol{\delta}^k$$

$$\boldsymbol{Q}^{k+1} = \boldsymbol{Q}^k - \frac{\boldsymbol{\delta}^k(\gamma^k)^T\boldsymbol{Q}^k + \boldsymbol{Q}^k\gamma^k(\boldsymbol{\delta}^{\boldsymbol{k}})^T}{(\boldsymbol{\delta}^k)^T\gamma^k} + \left(1 + \frac{(\gamma^k)^T\boldsymbol{Q}^k\gamma^k}{(\boldsymbol{\delta}^k)^T\boldsymbol{Q}^k}\right)\frac{\boldsymbol{\delta}^k(\boldsymbol{\delta}^k)^T}{(\boldsymbol{\delta}^k)^T\gamma^k}$$

BFGS does better than DFP with approximate line search.
Give an example where the BFGS update does not exist.
What would you do in this case?

The Broyden Class

Comparing the two solutions from DFP and BFGS,

$$\begin{cases} \text{DFP:} \quad \boldsymbol{Q}^{k+1} & = \boldsymbol{Q}^k - \frac{\boldsymbol{Q}^k \gamma^k (\gamma^k)^T \boldsymbol{Q}^k}{(\gamma^k)^T \boldsymbol{Q}^k \gamma^k} + \frac{\boldsymbol{\delta}(\boldsymbol{\delta}^k)^T}{(\boldsymbol{\delta}^k)^T \gamma^k} \\ \text{BFGS:} \; \boldsymbol{Q}^{k+1} & = \boldsymbol{Q}^k - \frac{\boldsymbol{\delta}^k (\gamma^k)^T \boldsymbol{Q}^k + \boldsymbol{Q}^k \gamma^k (\boldsymbol{\delta}^{\boldsymbol{k}})^T}{(\boldsymbol{\delta}^k)^T \gamma^k} + \left(1 + \frac{(\gamma^k)^T \boldsymbol{Q}^k \gamma^k}{(\boldsymbol{\delta}^k)^T \boldsymbol{Q}^k}\right) \frac{\boldsymbol{\delta}^k (\boldsymbol{\delta}^k)^T}{(\boldsymbol{\delta}^k)^T \gamma^k} \\ & = \boldsymbol{Q}^k - \frac{\boldsymbol{Q}^k \gamma^k (\gamma^k)^T \boldsymbol{Q}^k}{(\gamma^k)^T \boldsymbol{Q}^k \gamma^k} + \frac{\boldsymbol{\delta}(\boldsymbol{\delta}^k)^T}{(\boldsymbol{\delta}^k)^T \gamma^k} + [(\boldsymbol{\delta}^k)^T \boldsymbol{Q}^k \boldsymbol{\delta}^k] \boldsymbol{v}^{\boldsymbol{k}} (\boldsymbol{v}^{\boldsymbol{k}})^T \end{cases}$$

where

$$\boldsymbol{v}^{\boldsymbol{k}} = \frac{\gamma^k}{(\gamma^k)^T \boldsymbol{\delta}^k} - \frac{\boldsymbol{Q}^k \boldsymbol{\delta}^k}{(\boldsymbol{\delta}^k)^T \boldsymbol{Q}^k \boldsymbol{\delta}^k}$$

In fact, there exists a family of solutions

$$\boldsymbol{Q}^k = (1 - \lambda) \boldsymbol{Q}^k_{\text{BFGS}} + \lambda \boldsymbol{Q}^k_{\text{DFP}}$$

Changing λ from 0 to 1 is actually varying the u and v in

$$\boldsymbol{Q}^{k+1} = \boldsymbol{Q}^k + a u u^T + b v v^T$$

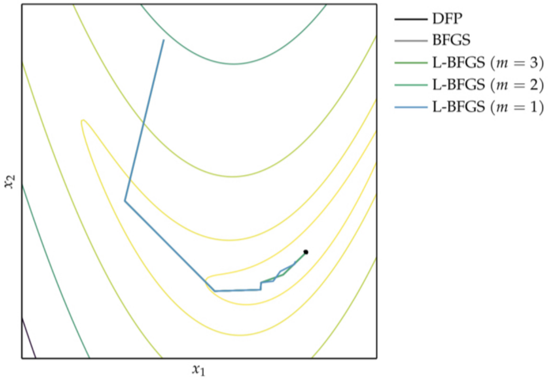
The Limited-memory BFGS

BFGS still uses an $n \times n$ dense matrix, which is a problem for storage of the hessian when dealing with very large scale problems. Inspecting the following update, can we do something smart?

$$\boldsymbol{Q}^{k+1} = \boldsymbol{Q}^k + auu^T + bvv^T$$

In L-BFGS, it stores the last m values for $\boldsymbol{\delta}$ and γ rather than the entire inverse of H.

Comparison of various quasi-Newton algorithms



Summary

- Quasi-Newton method attempted to approximate the Hessian from function and gradient evaluations.
- The first step approximation of hessian in the quasi-newton methods is usually an identity matrix
- BFGS performs better than DFP, but it still relies on the storage of big Hessian matrix
- L-BFGS is a more scalable approach for large scale problems.
- All quasi-Newton methods can work with the approximate line search.