Notes on Nocedal and Wright's "Numerical Optimization"
Chapter 8 – "Quasi-Newton Methods"

Lucas N. Ribeiro

Introduction I

- First ideas: WC Davidon at Argonne National Lab in the 1950s
- Requires only gradient knowledge to achieve super-linear convergence
- Sometimes more efficient than Newton's method because it does not require 2nd order derivatives

BFGS I

Highlights

- BFGS iterates as $x_{k+1} = x_k + \alpha_k p_k$, where p_k is obtained by minimizing a quadratic model at x_k, $p_k = -B_k^{-1} p_k$.
- Instead of recalculating a fresh B_k at each step, it updates in a simple manner.
- The update is the solution of the *secant equation*, which has solutions when the *curvature condition* is satisfied
- The BFGS updates B_k with a rank-2 matrix at each iteration
- Super-linear convergence

Consider the quadratic model and its gradient

$$m_k(p) = f_k + \nabla f_k^{\mathsf{T}} p + 0.5 p^{\mathsf{T}} B_k p, \quad \nabla m_k(p) = \nabla f_k + B_k p.$$

BFGS II

The secant equation

- From the update formula, we define $s_k = x_{k+1} - x_k = B_{k+1}(\alpha_k p_k)$.
- A reasonable condition for BFGS is that the gradient of m_{k+1} should match the gradient of f at the latest two iterates x_k and x_{k+1}. We have:

$$\nabla m_{k+1}(-\alpha_k p_k) = \nabla f_{k+1} - \alpha_k B_{k+1} p_k \overset{!}{=} \nabla f_k$$

- Therefore:

$$B_{k+1}(\alpha_k p_k) = \nabla f_{k+1} - \nabla f_k$$

  Defining $y_k = \nabla f_{k+1} - \nabla f_k$, we have the secant equation:

$$B_{k+1} s_k = y_k.$$

BFGS III

Solving the secant equations

- We wish to solve the secant equation $B_{k+1}s_k = y_k$ to nicely update our line search.
- Solving this system will be possible only if the curvature condition $s_k^\top y_k > 0$ (because then B_{k+1} will be positive definite)
- When f is strongly convex, then it is always satisfied.
- Otherwise, one has to be careful to enforce this condition on line search

When the curvature condition is satisfied, the system has in fact infinite solution. To find a single one, we impose additional conditions.

BFGS IV

We consider the following problem:

$$\min_{B} \|B - B_k\| \tag{1a}$$

$$\text{subject to} \quad B = B^{\mathsf{T}}, \quad Bs_k = y_k \tag{1b}$$

- The norm in this problem may be whatever. The weighted Frobenius norm gives an easy solution considering the average Hessian weight matrix.
- In this case, the unique solution to this problem gives the DFP formula:

$$B_{k+1} = (I - \gamma_k y_k s_k^{\mathsf{T}}) B_k (I - \gamma_k s_k y_k^{\mathsf{T}}) + \gamma_k y_k y_k^{\mathsf{T}} \tag{2}$$

  where $\gamma_k = 1/(y_k^{\mathsf{T}} s_k)$.
- Note that in order to calculate the step $p_k = -B_k^{-1} p_k$, we need the inverse of B_k.

BFGS V

- Define $H_k = B_k^{-1}$. Applying the Sherman-Morrison-Woodbury formula to H_k gives:

$$H_{k+1} = H_k - \frac{H_k y_k y_k^{\mathsf{T}} H_k}{y_k^{\mathsf{T}} H_k y_k} + \frac{s_k s_k^{\mathsf{T}}}{y_k^{\mathsf{T}} s_k}$$

- It's a rank-2 update!
- The DFP formula is effective, but was superseded by the BFGS formula

BFGS VI

- The BFGS formula is obtained by reformulating the secant equation as

$$H_{k+1} y_k = s_k \tag{3}$$

- (Note that we just left-multiplied the old version by H_{k+1})
- To obtain a unique solution, we solve

$$\min_H \|H - H_k\| \tag{4a}$$

$$\text{subject to} \quad H = H^\mathsf{T}, \quad H y_k = s_k \tag{4b}$$

- and the solution is:

$$H_{k+1} = (I - \rho_k s_k y_k^\mathsf{T}) H_k (I - \rho_k y_k s_k^\mathsf{T}) + \rho_k s_k s_k^\mathsf{T} \tag{5}$$

where $\rho_k = 1/(y_k^\mathsf{T} s_k)$.

BFGS VII

BFGS

1. Given: starting point x_0, conv. threshold ϵ and inverse Hessian approx. H_0 (identity matrix, for example)
2. $k \leftarrow 0$
3. While $\|\nabla f_k\| > \epsilon$
   - Compute search direction $p_k = -H_k \nabla f_k$
   - Update step $x_{k+1} = x_k + \alpha p_k$
   - Compute H_{k+1} by means of the BFGS formula

The SR1 method I

- The symmetric rank-1 update has the form

$$B_{k+1} = B_k + \sigma v v^\mathsf{T}$$

  where $\sigma \in \{-1, 1\}$ and v is chosen so that B_{k+1} satisfies the secant equation.

- With some algebra, we have that:

$$B_{k+1} = B_k + \frac{(y_k - B_k s_k)(y_k - B_k s_k)^\mathsf{T}}{(y_k - B_k s_k)^\mathsf{T} s_k}$$

- Applying the Sherman-Morrison formula:

$$H_{k+1} = H_k + \frac{(s_k - H_k y_k)(s_k - H_k y_k)^\mathsf{T}}{(s_k - H_k y_k)^\mathsf{T} y_k}$$

The SR1 method II

- The major drawback of SR1 concerns the denominator of the update term. If it vanishes, the algorithm may become unstable
- Despite of that, there are some cases when the curvature condition does not hold and BFGS does not work, which SR1 can work nicely.
- Trust-region implementations of SR1 and methods to prevent it from breaking down may turn this solution very effective

The Broyden class I

- Broyden-class updating formulae follow the general formula:

$$B_{k+1} = B_k - \frac{B_k s_k s_k^{\mathsf{T}} B_k}{s_k^{\mathsf{T}} B_k s_k} + \frac{y_k y_k^{\mathsf{T}}}{y_k^{\mathsf{T}} s_k} + \phi_k (s_k^{\mathsf{T}} B_k s_k) v_k v_k^{\mathsf{T}}$$

  where

$$v_k = \left[\frac{y_k}{y_k^{\mathsf{T}} s_k} - \frac{B_k s_k}{s_k^{\mathsf{T}} B_k s_k} \right]$$

- BFGS: $\phi_k = 0$, DFP: $\phi_k = 1$ and SR1: $\phi_k = s_k^{\mathsf{T}} y_k / (s_k^{\mathsf{T}} y_k - s_k^{\mathsf{T}} B_k s_k)$.
- Broyden class algorithms has remarkable properties when applied with exact line searches