Chapter 11 Quasi-Newton Methods
Introduction

- In Newton’s method, for a general nonlinear objective function, convergence to a solution cannot be guaranteed from an arbitrary initial point $x^{(0)}$.

- The idea behind Newton’s method is to locally approximate the function $f$ being minimized, at every iteration, by a quadratic function. The minimizer for the quadratic approximation is used as the starting point for the next iteration.

$$x^{(k+1)} = x^{(k)} - F(x^{(k)})^{-1}g^{(k)}$$

- Guarantee that the algorithm has the descent property by modifying as follows

$$x^{(k+1)} = x^{(k)} - \alpha_k F(x^{(k)})^{-1}g^{(k)}$$

where $\alpha_k$ is chosen to ensure that

$$f(x^{(k+1)}) < f(x^{(k)})$$
For example, we may choose $\alpha_k = \arg \min_{\alpha \geq 0} f(x^{(k)} - \alpha_k F(x^{(k)})^{-1} g^{(k)})$. We can then determine an appropriate value of $\alpha_k$ by performing a line search in the direction $-F(x^{(k)})^{-1} g^{(k)}$. Note that although the line search is simply the minimization of the real variable function $\phi_k(\alpha) = f(x^{(k)} - \alpha_k F(x^{(k)})^{-1} g^{(k)})$, it is not a trivial problem to solve.

A computational drawback of Newton’s method is the need to evaluate $F(x^{(k)})$ and solve the equation $F(x^{(k)}) d^{(k)} = -g^{(k)}$. To avoid the computation of $F(x^{(k)})^{-1}$, the quasi-Newton methods use an approximation to $F(x^{(k)})^{-1}$ in place of the true inverse.
Consider the formula

\[ x^{(k+1)} = x^{(k)} - \alpha H_k g^{(k)} \]

where \( H_k \) is an \( n \times n \) matrix and \( \alpha \) is a positive search parameter. Expanding \( f \) about \( x^{(k)} \) yields

\[ f(x^{(k+1)}) = f(x^{(k)}) + g^{(k)T}(x^{(k+1)} - x^{(k)}) + o(\|x^{(k+1)} - x^{(k)}\|) \]

\[ = f(x^{(k)}) - \alpha g^{(k)T} H_k g^{(k)} + o(\|H_k g^{(k)}\|/\alpha) \]

As \( \alpha \) tends to zero, the second term on the right-hand side dominates the third. Thus, to guarantee a decrease in \( f \) for small \( \alpha \), we have to have

\[ g^{(k)T} H_k g^{(k)} > 0 \]

A simple way to ensure this is to require that \( H_k \) be positive definite.
Proposition 11.1: Let $f \in C^1$, $x^{(k)} \in \mathbb{R}^n$, $g^{(k)} = \nabla f(x^{(k)}) \neq 0$, and $H_k$ an $n \times n$ real symmetric positive definite matrix. If we set $x^{(k+1)} = x^{(k)} - \alpha H_k g^{(k)}$, where $\alpha_k = \operatorname{arg\,min}_{\alpha \geq 0} f(x^{(k)} - \alpha H_k g^{(k)})$, then $\alpha_k > 0$ and $f(x^{(k+1)}) < f(x^{(k)})$. 
Approximating the Inverse Hessian

Let $H_0, H_1, H_2, \ldots$ be successive approximations of the inverse $F(x^{(k)})^{-1}$ of the Hessian.

Suppose first that the Hessian matrix $F(x)$ of the objective function $f$ is constant and independent of $x$. In other words, the objective function is quadratic, with Hessian $F(x) = Q$ for all $x$, where $Q = Q^T$. Then,

$$g^{(k+1)} - g^{(k)} = Q(x^{(k+1)} - x^{(k)})$$

Let

$$\Delta g^{(k)} \triangleq g^{(k+1)} - g^{(k)}
\Delta x^{(k)} \triangleq x^{(k+1)} - x^{(k)}$$

Then, we may write

$$\Delta g^{(k)} = Q\Delta x^{(k)}$$
Approximating the Inverse Hessian

- We start with a real symmetric positive definite matrix $H_0$. Note that given $k$, the matrix $Q^{-1}$ satisfies
  \[ Q^{-1} \Delta g^{(i)} = \Delta x^{(i)} \quad 0 \leq i \leq k \]

- Therefore, we also impose the requirement that the approximation $H_{k+1}$ of the Hessian satisfy
  \[ H_{k+1} \Delta g^{(i)} = \Delta x^{(i)} \quad 0 \leq i \leq k \]

- If $n$ steps are involved, then moving in $n$ directions $\Delta x^{(0)}, \Delta x^{(1)}, ..., \Delta x^{(n-1)}$ yields
  \[ H_n \Delta g^{(0)} = \Delta x^{(0)} \]
  \[ H_n \Delta g^{(1)} = \Delta x^{(1)} \]
  \[ \vdots \]
  \[ H_n \Delta g^{(n-1)} = \Delta x^{(n-1)} \]
Approximating the Inverse Hessian

This set of equations can be represented as

\[ H_n[\Delta g^{(0)}, \Delta g^{(1)}, ..., \Delta g^{(n-1)}] = [\Delta x^{(0)}, \Delta x^{(1)}, ..., \Delta x^{(n-1)}]. \]

Note that \( Q \) satisfies

\[ Q[\Delta x^{(0)}, \Delta x^{(1)}, ..., \Delta x^{(n-1)}] = [\Delta g^{(0)}, \Delta g^{(1)}, ..., \Delta g^{(n-1)}] \]

and

\[ Q^{-1}[\Delta g^{(0)}, \Delta g^{(1)}, ..., \Delta g^{(n-1)}] = [\Delta x^{(0)}, \Delta x^{(1)}, ..., \Delta x^{(n-1)}] \]

Therefore, if \( [\Delta g^{(0)}, \Delta g^{(1)}, ..., \Delta g^{(n-1)}] \) is nonsingular, then \( Q^{-1} \) is determined uniquely after \( n \) steps, via

\[ Q^{-1} = H_n = [\Delta x^{(0)}, \Delta x^{(1)}, ..., \Delta x^{(n-1)}][\Delta g^{(0)}, \Delta g^{(1)}, ..., \Delta g^{(n-1)}]^{-1} \]
Approximating the Inverse Hessian

- We conclude that if $H_n$ satisfies the equations
  \[ H_n \Delta g^{(i)} = \Delta x^{(i)}, \quad 0 \leq i \leq n - 1 \]
  then the algorithm
  \[ x^{(k+1)} = x^{(k)} - \alpha_k H_k g^{(k)}, \]
  \[ \alpha_k = \arg\min_{\alpha \geq 0} f(x^{(k)} - \alpha H_k g^{(k)}) \]
  is guaranteed to solve problems with quadratic objective functions in $n + 1$ steps, because the update
  \[ x^{(n+1)} = x^{(n)} - \alpha_n H_n g^{(n)} \]
  is equivalent to Newton’s algorithm.
Approximating the Inverse Hessian

- The quasi-Newton algorithms have the form

\[
\begin{align*}
  d^{(k)} &= -H_k g^{(k)} \\
  \alpha_k &= \text{arg min}_{\alpha \geq 0} f(x^{(k)} + \alpha d^{(k)}) \\
  x^{(k+1)} &= x^{(k)} + \alpha_k d^{(k)}
\end{align*}
\]

where the matrices $H_0, H_1, \ldots$ are symmetric. In the quadratic case these matrices are required to satisfy

\[ H_{k+1} \Delta g^{(i)} = \Delta x^{(i)}, \quad 0 \leq i \leq k \]

where $\Delta x^{(i)} = x^{(i+1)} - x^{(i)} = \alpha_i d^{(i)}$ and $\Delta g^{(i)} = g^{(i+1)} - g^{(i)} = Q \Delta x^{(i)}$

It turns out that quasi-Newton methods are also conjugate direction methods.
Theorem 11.1: Consider a quasi-Newton algorithm applied to a quadratic function with Hessian $Q = Q^T$ such that for $0 \leq k < n - 1$

$$H_{k+1} \Delta g^{(i)} = \Delta x^{(i)}, \quad 0 \leq i \leq k$$

where $H_{k+1} = H_{k+1}^T$. If $\alpha_i \neq 0$, $0 \leq i \leq k$, then $d^{(0)}, \ldots, d^{(k+1)}$ are $Q$-conjugate.

Proof: We proceed by induction. We begin with the $k = 0$ case: that $d^{(0)}$ and $d^{(1)}$ are $Q$-conjugate. Because $\alpha_0 \neq 0$, we can write $d^{(0)} = \Delta x^{(0)}/\alpha_0$. Hence, but $g^{(1)T}d^{(0)} = 0$ as a consequence of $\alpha_0 > 0$ being the minimizer of

$$\phi(\alpha) = f(x^{(0)} + \alpha d^{(0)})$$. Hence,

$$d^{(1)T}Qd^{(0)} = 0$$

Thus,

$$d^{(1)T}Qd^{(0)} = -g^{(1)T}H_1Qd^{(0)}$$

$$= -g^{(1)T}H_1Q\frac{\Delta x^{(0)}}{\alpha_0}$$

$$= -g^{(1)T}H_1\frac{\Delta g^{(0)}}{\alpha_0}$$

$$= -g^{(1)T}\frac{\Delta x^{(0)}}{\alpha_0}$$

$$= -g^{(1)T}d^{(0)}$$
Approximating the Inverse Hessian

Assume that the result is true for $k - 1$. We now prove that the result for $k$, that is, that $d^{(0)}, ..., d^{(k+1)}$ are $Q$-conjugate. If suffices to show that $d^{(k+1)^T}Qd^{(i)} = 0, 0 \leq i \leq k$. Given $0 \leq i \leq k$ using the same algebraic steps as in the $k = 0$ case, and using the assumption that $\alpha_i \neq 0$, we obtain

\[
\begin{align*}
  d^{(k+1)^T}Qd^{(i)} &= -g^{(k+1)^T}H_{k+1}Qd^{(i)} \\
  \vdots &= -g^{(k+1)^T}d^{(i)}
\end{align*}
\]

Because $d^{(0)}, ..., d^{(k)}$ are $Q$-conjugate by assumption, we conclude from Lemma 10.2 that $g^{(k+1)^T}d^{(i)} = 0$. Hence, $d^{(k+1)^T}Qd^{(i)} = 0$, which completes the proof.
The Rank One Correction Formula

- In the **rank one correction formula**, the correction term is symmetric and has the form \( a_k z^{(k)} z^{(k)T} \), where \( a_k \in R \) and \( z^{(k)} \in R^n \)
  The update equation is
  \[
  H_{k+1} = H_k + a_k z^{(k)} z^{(k)T}
  \]

  Note that
  \[
  \text{rank}(z^{(k)} z^{(k)T}) = \text{rank}\left(\begin{bmatrix} z_1^{(k)} \\ \vdots \\ z_n^{(k)} \end{bmatrix}\begin{bmatrix} z_1^{(k)} & \cdots & z_n^{(k)} \end{bmatrix}\right) = 1
  \]

  and hence the name **rank one correction** [also called **single-rank symmetric** (SRF) algorithm].
  The product \( z^{(k)} z^{(k)T} \) is sometimes referred to as the **dyadic product** or **outer product**. Observe that if \( H_k \) is symmetric, then so is \( H_{k+1} \)
Our goal now is to determine $a_k$ and $z^{(k)}$, given $H_k$, $\Delta g^{(k)}$, $\Delta x^{(k)}$, so that the required relationship discussed in Section 11.2 is satisfied; namely $H_{k+1}\Delta g^{(i)} = \Delta x^{(i)}$, $i = 1, \ldots, k$.

To begin, consider the condition $H_{k+1}\Delta g^{(k)} = \Delta x^{(k)}$. In other words, given $H_k$, $\Delta g^{(k)}$, $\Delta x^{(k)}$, we wish to find $a_k$ and $z^{(k)}$ to ensure that

$$H_{k+1}\Delta g^{(k)} = (H_k + a_k z^{(k)} z^{(k)T})\Delta g^{(k)} = \Delta x^{(k)}$$

First note that $z^{(k)T}\Delta g^{(k)}$ is a scalar. Thus,

$$\Delta x^{(k)} - H_k \Delta g^{(k)} = (a_k z^{(k)T} \Delta g^{(k)}) z^{(k)}$$

and hence

$$z^{(k)} = \frac{\Delta x^{(k)} - H_k \Delta g^{(k)}}{a_k (z^{(k)T} \Delta g^{(k)})}$$
The Rank One Correction Formula

- We can now determine

\[ a_k z^{(k)} z^{(k)T} = \frac{(\Delta x^{(k)} - H_k \Delta g^{(k)}) (\Delta x^{(k)} - H_k \Delta g^{(k)})^T}{a_k (z^{(k)T} \Delta g^{(k)})^2} \]

Hence,

\[ H_{k+1} = H^{(k)} + \frac{(\Delta x^{(k)} - H_k \Delta g^{(k)}) (\Delta x^{(k)} - H_k \Delta g^{(k)})^T}{a_k (z^{(k)T} \Delta g^{(k)})^2} \]

- The next step is to express the denominator of the second term on the right-hand side as a function of the given quantities

\[ H_k, \Delta g^{(k)}, \Delta x^{(k)} \]. Premultiply \( \Delta x^{(k)} - H_k \Delta g^{(k)} = (a_k z^{(k)T} \Delta g^{(k)}) z^{(k)} \) by \( \Delta g^{(k)T} \) to obtain

\[ \Delta g^{(k)T} \Delta x^{(k)} - \Delta g^{(k)T} H_k \Delta g^{(k)} = \Delta g^{(k)T} a_k z^{(k)} z^{(k)T} \Delta g^{(k)} \]
Observe that $a_k$ is a scalar and so is $\Delta g^{(k)T}z^{(k)} = z^{(k)T}\Delta g^{(k)}$. Thus,

$$\Delta g^{(k)T}\Delta x^{(k)} - \Delta g^{(k)T}H_k\Delta g^{(k)} = a_k(z^{(k)T}\Delta g^{(k)})^2$$

Taking this relation into account yields

$$H_{k+1} = H_k + \frac{(\Delta x^{(k)} - H_k\Delta g^{(k)})(\Delta x^{(k)} - H_k\Delta g^{(k)})^T}{\Delta g^{(k)T}(\Delta x^{(k)} - H_k\Delta g^{(k)})}$$
Rank One Algorithm

1. Set \( k := 0 \); select \( x^{(0)} \) and a real symmetric positive definite \( H_0 \).
2. If \( g^{(k)} = 0 \), stop; else, \( d^{(k)} = -H_k g^{(k)} \).
3. Compute
   \[
   \alpha_k = \arg \min_{\alpha \geq 0} f(x^{(k)} + \alpha d^{(k)})
   \]
   \[
   x^{(k+1)} = x^{(k)} + \alpha_k d^{(k)}
   \]
4. Compute
   \[
   \Delta x^{(k)} = \alpha_k d^{(k)}
   \]
   \[
   \Delta g^{(k)} = g^{(k+1)} - g^{(k)}
   \]
   \[
   H_{k+1} = H_k + \frac{(\Delta x^{(k)} - H_k \Delta g^{(k)}) (\Delta x^{(k)} - H_k \Delta g^{(k)})^T}{\Delta g^{(k)T} (\Delta x^{(k)} - H_k \Delta g^{(k)})}
   \]
5. Set \( k := k + 1 \); go to step 2.
Rank One Algorithm

- However, what we want is $H_{k+1} \Delta g^{(i)} = \Delta x^{(i)}, \; i = 1, \ldots, k$

- Theorem 11.2: For the rank one algorithm applied to the quadratic with Hessian $Q = Q^T$, we have $H_{k+1} \Delta g^{(i)} = \Delta x^{(i)}$

  \[ 0 \leq i \leq k \]

- Proof.
Example

- Let $f(x_1, x_2) = x_1^2 + \frac{1}{2}x_2^2 + 3$. Apply the rank one correction algorithm to minimize $f$. Use $x^{(0)} = [1, 2]^T$ and $H_0 = I_2$

- We can represent $f$ as

$$f(x) = \frac{1}{2}x^T \begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix} x + 3$$

Thus,

$$g^{(k)} = \begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix} x^{(k)}$$

Because $H_0 = I_2$, $d^{(0)} = -g^{(0)} = [-2, -2]^T$
Example

The objective function is quadratic, and hence

$$\alpha_0 = \arg \min_{\alpha \geq 0} f(x^{(0)} + \alpha d^{(0)})$$

$$= -\frac{g^{(0)}T d^{(0)}}{d^{(0)T} Q d^{(0)}} = \frac{[2, 2]}{[2, 2]} \begin{bmatrix} 2 \\ 2 \end{bmatrix} = \frac{2}{3}$$

and thus $x^{(1)} = x^{(0)} + \alpha_0 d^{(0)} = \begin{bmatrix} -\frac{1}{3} \\ \frac{2}{3} \end{bmatrix}^T$

We then compute

$$\Delta x^{(0)} = \alpha_0 d^{(0)} = \begin{bmatrix} \frac{4}{3} \\ \frac{4}{3} \end{bmatrix}^T$$

$$g^{(1)} = Q x^{(1)} = \begin{bmatrix} \frac{2}{3} \\ \frac{2}{3} \end{bmatrix}^T$$

$$\Delta g^{(0)} = g^{(1)} - g^{(0)} = \begin{bmatrix} -\frac{8}{3} \\ -\frac{4}{3} \end{bmatrix}^T$$
Example

Because

\[ \Delta g^{(0)T}(\Delta x^{(0)} - H_0\Delta g^{(0)}) = \begin{bmatrix} -\frac{8}{3} & -\frac{4}{3} \end{bmatrix} \begin{bmatrix} \frac{4}{3} \\ 0 \end{bmatrix} = -\frac{32}{9} \]

We obtain

\[ H_1 = H_0 + \frac{(\Delta x^{(0)} - H_0\Delta g^{(0)})(\Delta x^{(0)} - H_0\Delta g^{(0)})^T}{\Delta g^{(0)T}(\Delta x^{(0)} - H_0\Delta g^{(0)})} = \begin{bmatrix} \frac{1}{2} & 0 \\ 0 & 1 \end{bmatrix} \]

Therefore,

\[ d^{(1)} = -H_1 g^{(1)} = \begin{bmatrix} \frac{1}{3} \\ -\frac{2}{3} \end{bmatrix}^T \]

\[ \alpha_1 = -\frac{g^{(1)T}d^{(1)}}{d^{(1)T}Qd^{(1)}} = 1 \]

We now compute \( x^{(2)} = x^{(1)} + \alpha_1 a^{(1)} = [0, 0]^T \)

Note that \( g^{(2)} = 0 \), and therefore \( x^{(2)} = x^* \). As expected, the algorithm solves the problem in two steps.

Note that the directions \( d^{(0)}, d^{(1)} \) are \( Q \)-conjugate, in accordance with Theorem 11.1.
The Rank One Correction Formula

- Unfortunately, the rank one correction algorithm is not very satisfactory for several reasons.
  - The matrix $H_{k+1}$ that the rank one algorithm generates may not be positive definite and thus $d^{(k+1)}$ may not be a descent direction. This happens even in the quadratic case.
  - If $\Delta g^{(k)}T(\Delta x^{(k)} - H_k \Delta g^{(k)})$ is close to zero, then there may be numerical problems in evaluating $H_{k+1}$.
- Fortunately, alternative algorithms have been developed for updating $H_k$. In particular, if we use a “rank two” update, then $H_k$ is guaranteed to be positive definite for all $k$, provided that the line search is exact.
The DFP Algorithm

- This algorithm was developed by Davidon (1959), Fletcher, and Powell (1963).
- The DFP algorithm is also known as the variable metric algorithm.

DFP Algorithm

1. Set $k := 0$; select $x^{(0)}$ and a real symmetric positive definite $H_0$
2. If $g^{(k)} = 0$, stop; else, $d^{(k)} = -H_k g^{(k)}$
3. Compute $\alpha_k = \arg\min_{\alpha \geq 0} f(x^{(k)} + \alpha d^{(k)})$
   $$x^{(k+1)} = x^{(k)} + \alpha_k d^{(k)}$$
4. Compute
   $$\Delta x^{(k)} = \alpha_k d^{(k)}$$
   $$\Delta g^{(k)} = g^{(k+1)} - g^{(k)}$$
   $$H_{k+1} = H_k + \frac{\Delta x^{(k)}\Delta x^{(k)T}}{\Delta x^{(k)T}\Delta g^{(k)}} - \frac{[H_k \Delta g^{(k)}][H_k \Delta g^{(k)}]^T}{\Delta g^{(k)T} H_k \Delta g^{(k)}}$$
5. Set $k := k + 1$; go to step 2.
Theorem 11.3: In the DFP algorithm applied to the quadratic with Hessian \( Q = Q^T \), we have \( H_{k+1} \Delta g^{(i)} = \Delta x^{(i)}, \ 0 \leq i \leq k \)

Theorem 11.4: Suppose that \( g^{(k)} \neq 0 \). In the DFP algorithm, if \( H_k \) is positive definite, then so is \( H_{k+1} \).
Example

- Locate the minimizer of $f(x) = \frac{1}{2} x^T \begin{bmatrix} 4 & 2 \\ 2 & 2 \end{bmatrix} x - x^T \begin{bmatrix} -1 \\ 1 \end{bmatrix}$, $x \in \mathbb{R}^2$

Use the initial point $x^{(0)} = [0, 0]^T$ and $H_0 = I_2$

- Note that in this case
  \[ g^{(k)} = \begin{bmatrix} 4 & 2 \\ 2 & 2 \end{bmatrix} x^{(k)} - \begin{bmatrix} -1 \\ 1 \end{bmatrix} \]

Hence, $g^{(0)} = [1, -1]^T$

\[ d^{(0)} = -H_0 g^{(0)} = -\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ -1 \end{bmatrix} = \begin{bmatrix} -1 \\ 1 \end{bmatrix} \]

Because $f$ is a quadratic function,

\[ \alpha_0 = \arg \min_{\alpha \geq 0} f(x^{(0)} + \alpha d^{(0)}) = -\frac{g^{(0)T} d^{(0)}}{d^{(0)T} Q d^{(0)}} = 1 \]
Example

- Therefore, \( x^{(1)} = x^{(0)} + \alpha_0 d^{(0)} = [-1, 1]^T \)
- We then compute \( \Delta x^{(0)} = x^{(1)} - x^{(0)} = [-1, 1]^T \)
  \[
  g^{(1)} = \begin{bmatrix} 4 & 2 \\ 2 & 2 \end{bmatrix} \begin{bmatrix} -1 \\ 1 \end{bmatrix} - \begin{bmatrix} -1 \\ 1 \end{bmatrix} = \begin{bmatrix} -1 \\ -1 \end{bmatrix}
  \]
  \[
  \Delta g^{(0)} = g^{(1)} - g^{(0)} = [-2, 0]^T
  \]
- Observe that
  \[
  \Delta x^{(0)} \Delta x^{(0)T} = \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}
  \]
  \[
  \Delta x^{(0)T} \Delta g^{(0)} = 2
  \]
  \[
  H_0 \Delta g^{(0)} = \begin{bmatrix} -2 \\ 0 \end{bmatrix}
  \]

Thus,
\[
(H_0 \Delta g^{(0)})(H_0 \Delta g^{(0)})^T = \begin{bmatrix} 4 & 0 \\ 0 & 0 \end{bmatrix}
\]
\[
\Delta g^{(0)T} H_0 \Delta g^{(0)} = 4
\]
Example

\[
H_1 = H_0 + \frac{\Delta x^{(0)} \Delta x^{(0)T}}{\Delta x^{(0)T} \Delta g^{(0)}} - \frac{[H_0 \Delta g^{(0)}][H_0 \Delta g^{(0)}]^T}{\Delta g^{(0)T} H_0 \Delta g^{(0)}}
\]

\[
= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} + \frac{1}{2} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} - \frac{1}{4} \begin{bmatrix} 4 & 0 \\ 0 & 0 \end{bmatrix}
\]

\[
= \begin{bmatrix} \frac{1}{2} & -\frac{1}{2} \\ -\frac{1}{2} & \frac{3}{2} \end{bmatrix}
\]

- We now compute \( d^{(1)} = -H_1 g^{(1)} = [0, 1]^T \) and

\[
\alpha_1 = \arg \min_{\alpha \geq 0} f(x^{(1)} + \alpha d^{(1)}) = -\frac{g^{(1)T} d^{(1)}}{d^{(1)T} Q d^{(1)}} = \frac{1}{2}
\]

Hence, \( x^{(2)} = x^{(1)} + \alpha_0 d^{(1)} = [-1, 3/2]^T = x^* \), because \( f \) is a quadratic function of two variables.

- Note that we have \( d^{(0)T} Q d^{(1)} = d^{(1)T} Q d^{(0)} = 0 \); that is, \( d^{(0)} \) and \( d^{(1)} \) are \( Q \)-conjugate directions.
The DFP Algorithm

- The DFP algorithm is superior to the rank one algorithm in that it preserves the positive definiteness of $H_k$.
- However, it turns out that in the case of larger nonquadratic problems the algorithm has the tendency of sometimes getting stuck. This phenomenon is attributed to $H_k$ becoming nearly singular.
The BFGS Algorithm

- Suggested by Broyden, Fletcher, Goldfarb, and Shanno.
- Recall that the updating formulas for the approximation of the inverse of the Hessian matrix were based on satisfying the equations

\[ H_{k+1} \Delta g^{(i)} = \Delta x^{(i)} \quad 0 \leq i \leq k \]

which were derived from \( \Delta g^{(i)} = Q \Delta x^{(i)} \), \( 0 \leq i \leq k \). We then formulated update formulas for the approximations to the inverse of the Hessian matrix \( Q^{-1} \).
- An alternative to approximating \( Q^{-1} \) is to approximate \( Q \) itself.
The BFGS Algorithm

- Let $B_k$ be our estimate of $Q$ at the $k$th step. We require $B_{k+1}$ to satisfy $\Delta g^{(i)} = B_{k+1} \Delta x^{(i)}$, $0 \leq i \leq k$.

- Notice that this set of equations is similar to the previous set of equations for $H_{k+1}$, the only difference being that the roles of $\Delta x^{(i)}$ and $\Delta g^{(i)}$ are interchanged.

- Given any update formula for $H_k$, a corresponding update formula for $B_k$ can be found by interchanging the roles of $B_k$ and $H_k$ and of $\Delta g^{(k)}$ and $\Delta x^{(k)}$. In particular, the BFGS update for $B_k$ corresponds to the DFP update for $H_k$. Formulas related in this way are said to be dual or complementary.
The BFGS Algorithm

- Recall that the DFP update for the approximation $H_k$ of the inverse Hessian is
  \[ H_{k+1}^{DFP} = H_k + \frac{\Delta x^{(k)} \Delta x^{(k)T}}{\Delta x^{(k)T} \Delta g^{(k)}} - \frac{[H_k \Delta g^{(k)}][H_k \Delta g^{(k)}]^T}{\Delta g^{(k)T} H_k \Delta g^{(k)}} \]

- Using the complementarity concept, we can easily obtain an update equation for the approximation $B_k$ of the Hessian
  \[ B_{k+1} = B_k + \frac{\Delta g^{(k)} \Delta g^{(k)T}}{\Delta g^{(k)T} \Delta x^{(k)}} - \frac{[B_k \Delta x^{(k)}][B_k \Delta x^{(k)}]^T}{\Delta x^{(k)T} B_k \Delta x^{(k)}} \]

- To obtain the BFGS update for the approximation of the inverse Hessian, we take the inverse of $B_{k+1}$ to obtain
  \[ H_{k+1}^{BFGS} = (B_{k+1})^{-1} = \left( B_k + \frac{\Delta g^{(k)} \Delta g^{(k)T}}{\Delta g^{(k)T} \Delta x^{(k)}} - \frac{[B_k \Delta x^{(k)}][B_k \Delta x^{(k)}]^T}{\Delta x^{(k)T} B_k \Delta x^{(k)}} \right)^{-1} \]
The BFGS Algorithm

- Lemma 11.1 *Sherman-Morrison formula*: Let $A$ be a nonsingular matrix. Let $u$ and $v$ be column vectors such that $1 + v^T Au \neq 0$. Then, $A + uv^T$ is nonsingular, and its inverse can be written in terms of $A^{-1}$ using the following formula:

$$
(A + uv^T)^{-1} = A^{-1} - \frac{(A^{-1}u)(v^T A^{-1})}{1 + v^T A^{-1} u}
$$

- From Lemma 11.1 it follows that if $A^{-1}$ is known, then the inverse of the matrix $A$ augmented by a rank one matrix can be obtained by a modification of the matrix $A^{-1}$. 
Applying Lemma 11.1 twice to $B_{k+1}$ yields

$$H_{k+1}^{BFGS} = H_k + \left( 1 + \frac{\Delta g^{(k)} H_k \Delta g^{(k)}}{\Delta g^{(k)} T \Delta x^{(k)}} \right) \frac{\Delta x^{(k)} \Delta x^{(k)} T}{\Delta x^{(k)} T \Delta g^{(k)}}$$

Recall that for the quadratic case the DFP algorithm satisfies $H_{k+1}^{DFP} \Delta g^{(i)} = x^{(i)}, 0 \leq i \leq k$. Therefore, the BFGS update for $B_k$ satisfies $B_{k+1} \Delta x^{(i)} = g^{(i)}, 0 \leq i \leq k$. By construction of the BFGS formula for $H_{k+1}^{BFGS}$, we conclude that $H_{k+1}^{BFGS} \Delta g^{(i)} = \Delta x^{(i)}, 0 \leq i \leq k$. Hence, the BFGS algorithm enjoys all the properties of quasi-Newton methods, including the conjugate directions property. Moreover, the BFGS algorithm also inherits the positive definiteness property of the DFP algorithm; that is, if $g^{(k)} \neq 0$ and $H_k > 0$, then $H_{k+1}^{BFGS} > 0$.
Example

- The BFGS formula is often far more efficient than the DFP formula.

- Use the BFGS method to minimize $f(x) = \frac{1}{2}x^TQx - x^Tb + \log(\pi)$

  $$Q = \begin{bmatrix} 5 & -3 \\ -3 & 2 \end{bmatrix} \quad b = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

- Take $H_0 = I_2$ and $x_0 = [0, 0]^T$. Verify that $H_2 = Q^{-1}$.

- We have $d^{(0)} = -g^{(0)} = -(Qx^{(0)} - b) = b = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$

  The objective function is a quadratic, and hence we can use the following formula to compute $\alpha_0$

  $$\alpha_0 = -\frac{g^{(0)T}d^{(0)}}{d^{(0)T}Qd^{(0)}} = \frac{1}{2}$$
Example

Therefore, \( x^{(1)} = x^{(0)} + \alpha_0 d^{(0)} = \begin{bmatrix} 0 \\ 1/2 \end{bmatrix} \)

To compute \( H_1 = H_1^{BFGS} \), we need the following quantities:

\[
\Delta x^{(0)} = x^{(1)} - x^{(0)} = \begin{bmatrix} 0 \\ 1/2 \end{bmatrix}
\]

\[
g^{(1)} = Qx^{(1)} - b = \begin{bmatrix} -3/2 \\ 0 \end{bmatrix}
\]

\[
\Delta g^{(0)} = g^{(1)} - g^{(0)} = \begin{bmatrix} -3/2 \\ 1 \end{bmatrix}
\]

Therefore,

\[
H_1 = H_0 + \left(1 + \frac{\Delta g^{(0)T} H_0 \Delta g^{(0)}}{\Delta g^{(0)T} \Delta x^{(0)}} \right) \frac{\Delta x^{(0)} \Delta x^{(0)T}}{\Delta x^{(0)T} \Delta g^{(0)}} - \frac{\Delta x^{(0)} \Delta g^{(0)T} H_0 + H_0 \Delta g^{(0)} \Delta x^{(0)T}}{\Delta g^{(0)T} \Delta x^{(0)}} = \begin{bmatrix} 1 & 3/2 \\ 3/2 & 11/4 \end{bmatrix}
\]
Example

- Hence, we have \( d^{(1)} = -H_1g^{(1)} = \begin{bmatrix} 3/2 \\ 9/4 \end{bmatrix} \)
  \[
  \alpha_1 = -\frac{g^{(1)T}d^{(1)}}{d^{(1)T}Qd^{(1)}} = 2
  \]
  Therefore,
  \[
  x^{(2)} = x^{(1)} + \alpha_1 d^{(1)} = \begin{bmatrix} 3 \\ 5 \end{bmatrix}
  \]
  - Because our objective function is a quadratic on \( \mathbb{R}^2 \), \( x^{(2)} \) is the minimizer. Notice that the gradient at \( x^{(2)} \) is 0; that is, \( g^{(2)} = 0 \)
Example

To verify that $H_2 = Q^{-1}$, we compute

$$
\Delta x^{(1)} = x^{(2)} - x^{(1)} = \begin{bmatrix} 3 \\ 9/2 \end{bmatrix} \\
\Delta g^{(1)} = g^{(2)} - g^{(1)} = \begin{bmatrix} 3/2 \\ 0 \end{bmatrix}
$$

$$
H_2 = H_1 + \left(1 + \frac{\Delta g^{(1)T} H_1 \Delta g^{(1)}}{\Delta g^{(1)T} \Delta x^{(1)}}\right) \frac{\Delta x^{(1)} \Delta x^{(1)T}}{\Delta x^{(1)T} \Delta g^{(1)}} - \frac{\Delta x^{(1)} \Delta g^{(1)T}}{\Delta g^{(1)T} \Delta x^{(1)}} H_1 + \frac{H_1 \Delta g^{(1)}}{\Delta g^{(1)T} \Delta x^{(1)}} \Delta x^{(1)T} = \begin{bmatrix} 2 & 3 \\ 3 & 5 \end{bmatrix}
$$

$$
\implies H_2 Q = Q H_2 = I_2 \implies H_2 = Q^{-1}
$$