Chapter 11 Quasi-Newton Methods

An Introduction to Optimization

Spring, 2014

Wei-Ta Chu

Introduction

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

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

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

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

where α_k is chosen to ensure that

$$f(\boldsymbol{x}^{(k+1)}) < f(\boldsymbol{x}^{(k)})$$

Introduction

- For example, we may choose $\alpha_k = \arg\min_{\alpha \geq 0} f(\boldsymbol{x}^{(k)} - \alpha_k \boldsymbol{F}(\boldsymbol{x}^{(k)})^{-1}\boldsymbol{g}^{(k)})$ We can then determine an appropriate value of α_k by performing a line search in the direction $-\boldsymbol{F}(\boldsymbol{x}^{(k)})^{-1}\boldsymbol{g}^{(k)}$. Note that although the line search is simply the minimization of the real variable function $\phi_k(\alpha) = f(\boldsymbol{x}^{(k)} - \alpha_k \boldsymbol{F}(\boldsymbol{x}^{(k)})^{-1}\boldsymbol{g}^{(k)})$, it is not a trivial problem to solve.
- A computational drawback of Newton's method is the need to evaluate $\boldsymbol{F}(\boldsymbol{x}^{(k)})$ and solve the equation $\boldsymbol{F}(\boldsymbol{x}^{(k)})\boldsymbol{d}^{(k)} = -\boldsymbol{g}^{(k)}$. To avoid the computation of $\boldsymbol{F}(\boldsymbol{x}^{(k)})^{-1}$, the quasi-Newton methods use an approximation to $\boldsymbol{F}(\boldsymbol{x}^{(k)})^{-1}$ in place of the true inverse.

Introduction

- Consider the formula

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

where $\boldsymbol{H}_k$ is an $n \times n$ matrix and α is a positive search parameter. Expanding f about $\boldsymbol{x}^{(k)}$ yields

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

$$= f(\boldsymbol{x}^{(k)}) - \alpha \boldsymbol{g}^{(k)T} \boldsymbol{H}_k \boldsymbol{g}^{(k)} + o(\|\boldsymbol{H}_k \boldsymbol{g}^{(k)}\|\alpha)$$

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

$$\boldsymbol{g}^{(k)T} \boldsymbol{H}_k \boldsymbol{g}^{(k)} > 0$$

A simple way to ensure this is to require that $\boldsymbol{H}_k$ be positive definite.

Introduction

- Proposition 11.1: Let $f \in \mathcal{C}^1$, $\boldsymbol{x}^{(k)} \in R^n$, $\boldsymbol{g}^{(k)} = \nabla f(\boldsymbol{x}^{(k)}) \neq \mathbf{0}$, and $\boldsymbol{H}_k$ an $n \times n$ real symmetric positive definite matrix. If we set $\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} - \alpha \boldsymbol{H}_k \boldsymbol{g}^{(k)}$, where $\alpha_k = \arg\min_{\alpha \geq 0} f(\boldsymbol{x}^{(k)} - \alpha \boldsymbol{H}_k \boldsymbol{g}^{(k)})$, then $\alpha_k > 0$ and $f(\boldsymbol{x}^{(k+1)}) < f(\boldsymbol{x}^{(k)})$

$$\boxed{\boldsymbol{g}^{(k)} = \nabla f(\boldsymbol{x}^{(k)}) \qquad \nabla f(\boldsymbol{x}) = \boldsymbol{Q}\boldsymbol{x} - \boldsymbol{b}}$$

Approximating the Inverse Hessian

- Let $\boldsymbol{H}_0, \boldsymbol{H}_1, \boldsymbol{H}_2, \ldots$ be successive approximations of the inverse $\boldsymbol{F}(\boldsymbol{x}^{(k)})^{-1}$ of the Hessian.
- Suppose first that the Hessian matrix $\boldsymbol{F}(\boldsymbol{x})$ of the objective function f is constant and independent of $\boldsymbol{x}$. In other words, the objective function is quadratic, with Hessian $\boldsymbol{F}(\boldsymbol{x}) = \boldsymbol{Q}$ for all $\boldsymbol{x}$, where $\boldsymbol{Q} = \boldsymbol{Q}^T$. Then,

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

Let

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

Then, we may write

$$\Delta\boldsymbol{g}^{(k)} = \boldsymbol{Q}\Delta\boldsymbol{x}^{(k)}$$

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

Approximating the Inverse Hessian

- We start with a real symmetric positive definite matrix $\boldsymbol{H}_0$. Note that given k, the matrix $\boldsymbol{Q}^{-1}$ satisfies

$$\boldsymbol{Q}^{-1}\Delta\boldsymbol{g}^{(i)} = \Delta\boldsymbol{x}^{(i)} \qquad 0 \le i \le k$$

- Therefore, we also impose the requirement that the approximation $\boldsymbol{H}_{k+1}$ of the Hessian satisfy

$$\boldsymbol{H}_{k+1}\Delta\boldsymbol{g}^{(i)} = \Delta\boldsymbol{x}^{(i)} \qquad 0 \le i \le k$$

- If n steps are involved, then moving in n directions $\Delta\boldsymbol{x}^{(0)}, \Delta\boldsymbol{x}^{(1)}, \ldots, \Delta\boldsymbol{x}^{(n-1)}$ yields

$$\begin{aligned}
\boldsymbol{H}_n\Delta\boldsymbol{g}^{(0)} &= \Delta\boldsymbol{x}^{(0)} \\
\boldsymbol{H}_n\Delta\boldsymbol{g}^{(1)} &= \Delta\boldsymbol{x}^{(1)} \\
&\vdots \\
\boldsymbol{H}_n\Delta\boldsymbol{g}^{(n-1)} &= \Delta\boldsymbol{x}^{(n-1)}
\end{aligned}$$

Approximating the Inverse Hessian

- This set of equations can be represented as

$$\boldsymbol{H}_n[\Delta\boldsymbol{g}^{(0)}, \Delta\boldsymbol{g}^{(1)}, ..., \Delta\boldsymbol{g}^{(n-1)}] = [\Delta\boldsymbol{x}^{(0)}, \Delta\boldsymbol{x}^{(1)}, ..., \Delta\boldsymbol{x}^{(n-1)}]$$

Note that $\boldsymbol{Q}$ satisfies

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

and

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

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

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

Approximating the Inverse Hessian

- We conclude that if $\boldsymbol{H}_n$ satisfies the equations
$$\boldsymbol{H}_n \Delta \boldsymbol{g}^{(i)} = \Delta \boldsymbol{x}^{(i)}, 0 \leq i \leq n-1$$
then the algorithm $\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} - \alpha_k \boldsymbol{H}_k \boldsymbol{g}^{(k)}$,
$\alpha_k = \arg\min_{\alpha \geq 0} f(\boldsymbol{x}^{(k)} - \alpha \boldsymbol{H}_k \boldsymbol{g}^{(k)})$, is guaranteed to solve problems with quadratic objective functions in $n+1$ steps, because the update $\boldsymbol{x}^{(n+1)} = \boldsymbol{x}^{(n)} - \alpha_n \boldsymbol{H}_n \boldsymbol{g}^{(n)}$ is equivalent to Newton's algorithm.

Approximating the Inverse Hessian

- The quasi-Newton algorithms have the form

$$\boxed{\begin{aligned} \boldsymbol{d}^{(k)} &= -\boldsymbol{H}_k \boldsymbol{g}^{(k)} \\ \alpha_k &= \arg\min_{\alpha \geq 0} f(\boldsymbol{x}^{(k)} + \alpha \boldsymbol{d}^{(k)}) \\ \boldsymbol{x}^{(k+1)} &= \boldsymbol{x}^{(k)} + \alpha_k \boldsymbol{d}^{(k)} \end{aligned}}$$

  where the matrices $\boldsymbol{H}_0, \boldsymbol{H}_1, \ldots$ are symmetric. In the quadratic case these matrices are required to satisfy

$$\boldsymbol{H}_{k+1} \Delta \boldsymbol{g}^{(i)} = \Delta \boldsymbol{x}^{(i)}, 0 \leq i \leq k$$

  where $\Delta \boldsymbol{x}^{(i)} = \boldsymbol{x}^{(i+1)} - \boldsymbol{x}^{(i)} = \alpha_i \boldsymbol{d}^{(i)}$ and $\Delta \boldsymbol{g}^{(i)} = \boldsymbol{g}^{(i+1)} - \boldsymbol{g}^{(i)} = \boldsymbol{Q} \Delta \boldsymbol{x}^{(i)}$
  It turns out that quasi-Newton methods are also conjugate direction methods.

Approximating the Inverse Hessian

- Theorem 11.1: Consider a quasi-Newton algorithm applied to a quadratic function with Hessian $\boldsymbol{Q} = \boldsymbol{Q}^T$ such that for $0 \le k < n-1$
$$\boldsymbol{H}_{k+1}\Delta\boldsymbol{g}^{(i)} = \Delta\boldsymbol{x}^{(i)}, 0 \le i \le k$$
where $\boldsymbol{H}_{k+1} = \boldsymbol{H}_{k+1}^T$. If $\alpha_i \neq 0$, $0 \le i \le k$, then $\boldsymbol{d}^{(0)}, \ldots, \boldsymbol{d}^{(k+1)}$ are $\boldsymbol{Q}$-conjugate.
- Proof: We proceed by induction. We begin with the $k=0$ case: that $\boldsymbol{d}^{(0)}$ and $\boldsymbol{d}^{(1)}$ are $\boldsymbol{Q}$-conjugate. Because $\alpha_0 \neq 0$, we can write $\boldsymbol{d}^{(0)} = \Delta\boldsymbol{x}^{(0)}/\alpha_0$. Hence,
$$\begin{aligned}\boldsymbol{d}^{(1)T}\boldsymbol{Q}\boldsymbol{d}^{(0)} &= -\boldsymbol{g}^{(1)T}\boldsymbol{H}_1\boldsymbol{Q}\boldsymbol{d}^{(0)}\\ &= -\boldsymbol{g}^{(1)T}\boldsymbol{H}_1\frac{\boldsymbol{Q}\Delta\boldsymbol{x}^{(0)}}{\alpha_0}\\ &= -\boldsymbol{g}^{(1)T}\frac{\boldsymbol{H}_1\Delta\boldsymbol{g}^{(0)}}{\alpha_0}\\ &= -\boldsymbol{g}^{(1)T}\frac{\Delta\boldsymbol{x}^{(0)}}{\alpha_0}\\ &= -\boldsymbol{g}^{(1)T}\boldsymbol{d}^{(0)}\end{aligned}$$
but $\boldsymbol{g}^{(1)T}\boldsymbol{d}^{(0)} = 0$ as a consequence of $\alpha_0 > 0$ being the minimizer of $\phi(\alpha) = f(\boldsymbol{x}^{(0)} + \alpha\boldsymbol{d}^{(0)})$. Hence,
$$\boldsymbol{d}^{(1)T}\boldsymbol{Q}\boldsymbol{d}^{(0)} = 0$$

Approximating the Inverse Hessian

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

$$\begin{aligned} \boldsymbol{d}^{(k+1)T}\boldsymbol{Q}\boldsymbol{d}^{(i)} &= -\boldsymbol{g}^{(k+1)T}\boldsymbol{H}_{k+1}\boldsymbol{Q}\boldsymbol{d}^{(i)} \\ &\vdots \\ &= -\boldsymbol{g}^{(k+1)T}\boldsymbol{d}^{(i)} \end{aligned}$$

  Because $\boldsymbol{d}^{(0)}, \ldots, \boldsymbol{d}^{(k)}$ are $\boldsymbol{Q}$-conjugate by assumption, we conclude from Lemma 10.2 that $\boldsymbol{g}^{(k+1)T}\boldsymbol{d}^{(i)} = 0$. Hence, $\boldsymbol{d}^{(k+1)T}\boldsymbol{Q}\boldsymbol{d}^{(i)} = 0$, which completes the proof.

The Rank One Correction Formula

- In the ***rank one correction formula***, the correction term is symmetric and has the form $a_k \boldsymbol{z}^{(k)} \boldsymbol{z}^{(k)T}$, where $a_k \in R$ and $\boldsymbol{z}^{(k)} \in R^n$
  The update equation is
  $$\boldsymbol{H}_{k+1} = \boldsymbol{H}_k + a_k \boldsymbol{z}^{(k)} \boldsymbol{z}^{(k)T}$$
  Note that
  $$\operatorname{rank}(\boldsymbol{z}^{(k)} \boldsymbol{z}^{(k)T}) = \operatorname{rank}\left(\begin{bmatrix} z_1^{(k)} \\ \vdots \\ z_n^{(n)} \end{bmatrix} \begin{bmatrix} z_1^{(k)} & \cdots & z_n^{(k)} \end{bmatrix} \right) = 1$$
  and hence the name ***rank one correction*** [also called ***single-rank symmetric*** (SRF) algorithm].
  The product $\boldsymbol{z}^{(k)} \boldsymbol{z}^{(k)T}$ is sometimes referred to as the ***dyadic product*** or ***outer product***. Observe that if $\boldsymbol{H}_k$ is symmetric, then so is $\boldsymbol{H}_{k+1}$

The Rank One Correction Formula

- Our goal now is to determine a_k and $\boldsymbol{z}^{(k)}$, given $\boldsymbol{H}_k$, $\Delta\boldsymbol{g}^{(k)}$, $\Delta\boldsymbol{x}^{(k)}$ so that the required relationship discussed in Section 11.2 is satisfied; namely $\boldsymbol{H}_{k+1}\Delta\boldsymbol{g}^{(i)} = \Delta\boldsymbol{x}^{(i)}, i = 1, \ldots, k$.
- To begin, consider the condition $\boldsymbol{H}_{k+1}\Delta\boldsymbol{g}^{(k)} = \Delta\boldsymbol{x}^{(k)}$. In other words, given $\boldsymbol{H}_k$, $\Delta\boldsymbol{g}^{(k)}$, $\Delta\boldsymbol{x}^{(k)}$, we wish to find a_k and $\boldsymbol{z}^{(k)}$ to ensure that

$$\boldsymbol{H}_{k+1}\Delta\boldsymbol{g}^{(k)} = (\boldsymbol{H}_k + a_k\boldsymbol{z}^{(k)}\boldsymbol{z}^{(k)T})\Delta\boldsymbol{g}^{(k)} = \Delta\boldsymbol{x}^{(k)}$$

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

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

and hence

$$\boldsymbol{z}^{(k)} = \frac{\Delta\boldsymbol{x}^{(k)} - \boldsymbol{H}_k\Delta\boldsymbol{g}^{(k)}}{a_k(\boldsymbol{z}^{(k)T}\Delta\boldsymbol{g}^{(k)})}$$

$$\boldsymbol{H}_{k+1}\Delta\boldsymbol{g}^{(k)} = (\boldsymbol{H}_k + a_k\boldsymbol{z}^{(k)}\boldsymbol{z}^{(k)T})\Delta\boldsymbol{g}^{(k)} = \Delta\boldsymbol{x}^{(k)}$$

The Rank One Correction Formula

- We can now determine

$$a_k\boldsymbol{z}^{(k)}\boldsymbol{z}^{(k)T} = \frac{(\Delta\boldsymbol{x}^{(k)} - \boldsymbol{H}_k\Delta\boldsymbol{g}^{(k)})(\Delta\boldsymbol{x}^{(k)} - \boldsymbol{H}_k\Delta\boldsymbol{g}^{(k)})^T}{a_k(\boldsymbol{z}^{(k)T}\Delta\boldsymbol{g}^{(k)})^2}$$

Hence,

$$\boldsymbol{H}_{k+1} = \boldsymbol{H}_{(k)} + \frac{(\Delta\boldsymbol{x}^{(k)} - \boldsymbol{H}_k\Delta\boldsymbol{g}^{(k)})(\Delta\boldsymbol{x}^{(k)} - \boldsymbol{H}_k\Delta\boldsymbol{g}^{(k)})^T}{a_k(\boldsymbol{z}^{(k)T}\Delta\boldsymbol{g}^{(k)})^2}$$

- The next step is to express the denominator of the second term on the right-hand side as a function of the given quantities $\boldsymbol{H}_k$, $\Delta\boldsymbol{g}^{(k)}$, $\Delta\boldsymbol{x}^{(k)}$. Premultiply $\Delta\boldsymbol{x}^{(k)} - \boldsymbol{H}_k\Delta\boldsymbol{g}^{(k)} = (a_k\boldsymbol{z}^{(k)T}\Delta\boldsymbol{g}^{(k)})\boldsymbol{z}^{(k)}$ by $\Delta\boldsymbol{g}^{(k)T}$ to obtain

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

$$\boxed{\Delta \boldsymbol{g}^{(k)T}\Delta \boldsymbol{x}^{(k)} - \Delta \boldsymbol{g}^{(k)T}\boldsymbol{H}_k\Delta \boldsymbol{g}^{(k)} = \Delta \boldsymbol{g}^{(k)T} a_k \boldsymbol{z}^{(k)}\boldsymbol{z}^{(k)T}\Delta \boldsymbol{g}^{(k)}}$$

The Rank One Correction Formula

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

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

Taking this relation into account yields

$$\boldsymbol{H}_{k+1} = \boldsymbol{H}_k + \frac{(\Delta \boldsymbol{x}^{(k)} - \boldsymbol{H}_k\Delta \boldsymbol{g}^{(k)})(\Delta \boldsymbol{x}^{(k)} - \boldsymbol{H}_k\Delta \boldsymbol{g}^{(k)})^T}{\Delta \boldsymbol{g}^{(k)T}(\Delta \boldsymbol{x}^{(k)} - \boldsymbol{H}_k\Delta \boldsymbol{g}^{(k)})}$$

$$\boxed{\boldsymbol{H}_{k+1} = \boldsymbol{H}_{(k)} + \frac{(\Delta \boldsymbol{x}^{(k)} - \boldsymbol{H}_k\Delta \boldsymbol{g}^{(k)})(\Delta \boldsymbol{x}^{(k)} - \boldsymbol{H}_k\Delta \boldsymbol{g}^{(k)})^T}{a_k(\boldsymbol{z}^{(k)T}\Delta \boldsymbol{g}^{(k)})^2}}$$

Rank One Algorithm

- 1. Set $k := 0$; select $\boldsymbol{x}^{(0)}$ and a real symmetric positive definite $\boldsymbol{H}_0$
- 2. If $\boldsymbol{g}^{(k)} = \mathbf{0}$, stop; else, $\boldsymbol{d}^{(k)} = -\boldsymbol{H}_k \boldsymbol{g}^{(k)}$
- 3. Compute

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

$$\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} + \alpha_k \boldsymbol{d}^{(k)}$$

- 4. Compute

$$\Delta \boldsymbol{x}^{(k)} = \alpha_k \boldsymbol{d}^{(k)}$$

$$\Delta \boldsymbol{g}^{(k)} = \boldsymbol{g}^{(k+1)} - \boldsymbol{g}^{(k)}$$

$$\boldsymbol{H}_{k+1} = \boldsymbol{H}_k + \frac{(\Delta \boldsymbol{x}^{(k)} - \boldsymbol{H}_k \Delta \boldsymbol{g}^{(k)})(\Delta \boldsymbol{x}^{(k)} - \boldsymbol{H}_k \Delta \boldsymbol{g}^{(k)})^T}{\Delta \boldsymbol{g}^{(k)T}(\Delta \boldsymbol{x}^{(k)} - \boldsymbol{H}_k \Delta \boldsymbol{g}^{(k)})}$$

- 5. Set $k := k + 1$; go to step 2.

Rank One Algorithm

- However, what we want is $\boldsymbol{H}_{k+1}\Delta\boldsymbol{g}^{(i)} = \Delta\boldsymbol{x}^{(i)}, i = 1, ..., k$
- Theorem 11.2: For the rank one algorithm applied to the quadratic with Hessian $\boldsymbol{Q} = \boldsymbol{Q}^T$, we have $\boldsymbol{H}_{k+1}\Delta\boldsymbol{g}^{(i)} = \Delta\boldsymbol{x}^{(i)}$ $0 \leq i \leq k$
- 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 $\boldsymbol{x}^{(0)} = [1, 2]^T$ and $\boldsymbol{H}_0 = \boldsymbol{I}_2$
- We can represent f as

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

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

Because $\boldsymbol{H}_0 = \boldsymbol{I}_2$, $\boldsymbol{d}^{(0)} = -\boldsymbol{g}^{(0)} = [-2, -2]^T$

Example

- The objective function is quadratic, and hence

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

$$= -\frac{\boldsymbol{g}^{(0)T}\boldsymbol{d}^{(0)}}{\boldsymbol{d}^{(0)T}\boldsymbol{Q}\boldsymbol{d}^{(0)}} = \frac{[2,2]\begin{bmatrix}2\\2\end{bmatrix}}{[2,2]\begin{bmatrix}2 & 0\\0 & 1\end{bmatrix}\begin{bmatrix}2\\2\end{bmatrix}} = \frac{2}{3}$$

and thus $\boldsymbol{x}^{(1)} = \boldsymbol{x}^{(0)} + \alpha_0 \boldsymbol{d}^{(0)} = [-\frac{1}{3}, \frac{2}{3}]^T$

We then compute

$$\Delta\boldsymbol{x}^{(0)} = \alpha_0 \boldsymbol{d}^{(0)} = [-\tfrac{4}{3}, -\tfrac{4}{3}]^T$$

$$\boldsymbol{g}^{(1)} = \boldsymbol{Q}\boldsymbol{x}^{(1)} = [-\tfrac{2}{3}, \tfrac{2}{3}]^T$$

$$\Delta\boldsymbol{g}^{(0)} = \boldsymbol{g}^{(1)} - \boldsymbol{g}^{(0)} = [-\tfrac{8}{3}, -\tfrac{4}{3}]^T$$

Example

- Because

$$\Delta \boldsymbol{g}^{(0)T}(\Delta \boldsymbol{x}^{(0)} - \boldsymbol{H}_0 \Delta \boldsymbol{g}^{(0)}) = [-\tfrac{8}{3}, -\tfrac{4}{3}] \begin{bmatrix} \frac{4}{3} \\ 0 \end{bmatrix} = -\frac{32}{9}$$

We obtain

$$\boldsymbol{H}_1 = \boldsymbol{H}_0 + \frac{(\Delta \boldsymbol{x}^{(0)} - \boldsymbol{H}_0 \Delta \boldsymbol{g}^{(0)})(\Delta \boldsymbol{x}^{(0)} - \boldsymbol{H}_0 \Delta \boldsymbol{g}^{(0)})^T}{\Delta \boldsymbol{g}^{(0)T}(\Delta \boldsymbol{x}^{(0)} - \boldsymbol{H}_0 \Delta \boldsymbol{g}^{(0)})} = \begin{bmatrix} \frac{1}{2} & 0 \\ 0 & 1 \end{bmatrix}$$

Therefore,

$$\boldsymbol{d}^{(1)} = -\boldsymbol{H}_1 \boldsymbol{g}^{(1)} = [\tfrac{1}{3}, -\tfrac{2}{3}]^T$$

$$\alpha_1 = -\frac{\boldsymbol{g}^{(1)T} \boldsymbol{d}^{(1)}}{\boldsymbol{d}^{(1)T} \boldsymbol{Q} \boldsymbol{d}^{(1)}} = 1$$

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

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

Note that the directions $\boldsymbol{d}^{(0)}, \boldsymbol{d}^{(1)}$ are $\boldsymbol{Q}$ -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 $\boldsymbol{H}_{k+1}$ that the rank one algorithm generates may not be positive definite and thus $\boldsymbol{d}^{(k+1)}$ may not be a descent direction. This happens even in the quadratic case.
  - If $\Delta \boldsymbol{g}^{(k)T}\left(\Delta \boldsymbol{x}^{(k)} - \boldsymbol{H}_k \Delta \boldsymbol{g}^{(k)}\right)$ is close to zero, then there may be numerical problems in evaluating $\boldsymbol{H}_{k+1}$.
- Fortunately, alternative algorithms have been developed for updating $\boldsymbol{H}_k$. In particular, if we use a “rank two” update, then $\boldsymbol{H}_k$ 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 $\boldsymbol{x}^{(0)}$ and a real symmetric positive definite $\boldsymbol{H}_0$
  - 2. If $\boldsymbol{g}^{(k)} = \boldsymbol{0}$, stop; else, $\boldsymbol{d}^{(k)} = -\boldsymbol{H}_k \boldsymbol{g}^{(k)}$
  - 3. Compute $\alpha_k = \arg\min_{\alpha \geq 0} f\left(\boldsymbol{x}^{(k)} + \alpha \boldsymbol{d}^{(k)}\right)$
    $$\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} + \alpha_k \boldsymbol{d}^{(k)}$$
  - 4. Compute
    $$\boxed{\begin{aligned} \Delta \boldsymbol{x}^{(k)} &= \alpha_k \boldsymbol{d}^{(k)} \\ \Delta \boldsymbol{g}^{(k)} &= \boldsymbol{g}^{(k+1)} - \boldsymbol{g}^{(k)} \\ \boldsymbol{H}_{k+1} &= \boldsymbol{H}_k + \frac{\Delta \boldsymbol{x}^{(k)} \Delta \boldsymbol{x}^{(k)T}}{\Delta \boldsymbol{x}^{(k)T} \Delta \boldsymbol{g}^{(k)}} - \frac{[\boldsymbol{H}_k \Delta \boldsymbol{g}^{(k)}][\boldsymbol{H}_k \Delta \boldsymbol{g}^{(k)}]^T}{\Delta \boldsymbol{g}^{(k)T} \boldsymbol{H}_k \Delta \boldsymbol{g}^{(k)}} \end{aligned}}$$
  - 5. Set $k := k + 1$; go to step 2.

The DFP Algorithm

- Theorem 11.3: In the DFP algorithm applied to the quadratic with Hessian $\boldsymbol{Q} = \boldsymbol{Q}^T$, we have $\boldsymbol{H}_{k+1}\Delta\boldsymbol{g}^{(i)} = \Delta\boldsymbol{x}^{(i)}$, $0 \leq i \leq k$
- Theorem 11.4: Suppose that $\boldsymbol{g}^{(k)} \neq \mathbf{0}$. In the DFP algorithm, if $\boldsymbol{H}_k$ is positive definite, then so is $\boldsymbol{H}_{k+1}$.

Example

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

  Use the initial point $\boldsymbol{x}^{(0)} = [0, 0]^T$ and $\boldsymbol{H}_0 = \boldsymbol{I}_2$
- Note that in this case

$$\boldsymbol{g}^{(k)} = \begin{bmatrix} 4 & 2 \\ 2 & 2 \end{bmatrix} \boldsymbol{x}^{(k)} - \begin{bmatrix} -1 \\ 1 \end{bmatrix}$$

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

$$\boldsymbol{d}^{(0)} = -\boldsymbol{H}_0 \boldsymbol{g}^{(0)} = -\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ -1 \end{bmatrix} = \begin{bmatrix} -1 \\ 1 \end{bmatrix}$$

  Because f is a quadratic function,

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

Example

- Therefore, $\boldsymbol{x}^{(1)} = \boldsymbol{x}^{(0)} + \alpha_0 \boldsymbol{d}^{(0)} = [-1, 1]^T$
- We then compute $\Delta \boldsymbol{x}^{(0)} = \boldsymbol{x}^{(1)} - \boldsymbol{x}^{(0)} = [-1, 1]^T$

$$\boldsymbol{g}^{(1)} = \begin{bmatrix} 4 & 2 \\ 2 & 2 \end{bmatrix} \begin{bmatrix} -1 \\ 1 \end{bmatrix} - \begin{bmatrix} -1 \\ 1 \end{bmatrix} = \begin{bmatrix} -1 \\ -1 \end{bmatrix}$$

$$\Delta \boldsymbol{g}^{(0)} = \boldsymbol{g}^{(1)} - \boldsymbol{g}^{(0)} = [-2, 0]^T$$

- Observe that

$$\Delta \boldsymbol{x}^{(0)} \Delta \boldsymbol{x}^{(0)T} = \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}$$

$$\Delta \boldsymbol{x}^{(0)T} \Delta \boldsymbol{g}^{(0)} = 2$$

$$\boldsymbol{H}_0 \Delta \boldsymbol{g}^{(0)} = \begin{bmatrix} -2 \\ 0 \end{bmatrix}$$

Thus,

$$(\boldsymbol{H}_0 \Delta \boldsymbol{g}^{(0)})(\boldsymbol{H}_0 \Delta \boldsymbol{g}^{(0)})^T = \begin{bmatrix} 4 & 0 \\ 0 & 0 \end{bmatrix} \qquad \Delta \boldsymbol{g}^{(0)T} \boldsymbol{H}_0 \Delta \boldsymbol{g}^{(0)} = 4$$

Example

$$\begin{aligned}\boldsymbol{H}_1 &= \boldsymbol{H}_0 + \frac{\Delta\boldsymbol{x}^{(0)}\Delta\boldsymbol{x}^{(0)T}}{\Delta\boldsymbol{x}^{(0)T}\Delta\boldsymbol{g}^{(0)}} - \frac{[\boldsymbol{H}_0\Delta\boldsymbol{g}^{(0)}][\boldsymbol{H}_0\Delta\boldsymbol{g}^{(0)}]^T}{\Delta\boldsymbol{g}^{(0)T}\boldsymbol{H}_0\Delta\boldsymbol{g}^{(0)}} \\ &= \begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix} + \frac{1}{2}\begin{bmatrix}1 & -1\\ -1 & 1\end{bmatrix} - \frac{1}{4}\begin{bmatrix}4 & 0\\ 0 & 0\end{bmatrix} \\ &= \begin{bmatrix}\frac{1}{2} & -\frac{1}{2}\\ -\frac{1}{2} & \frac{3}{2}\end{bmatrix}\end{aligned}$$

- We now compute $\boldsymbol{d}^{(1)} = -\boldsymbol{H}_1\boldsymbol{g}^{(1)} = [0,1]^T$ and

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

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

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

The DFP Algorithm

- The DFP algorithm is superior to the rank one algorithm in that it preserves the positive definiteness of $\boldsymbol{H}_k$.
- 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 $\boldsymbol{H}_k$ 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

$$\boldsymbol{H}_{k+1}\Delta\boldsymbol{g}^{(i)} = \Delta\boldsymbol{x}^{(i)} \qquad 0 \le i \le k$$

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

The BFGS Algorithm

- Let $\boldsymbol{B}_k$ be our estimate of $\boldsymbol{Q}$ at the kth step. We require $\boldsymbol{B}_{k+1}$ to satisfy $\Delta \boldsymbol{g}^{(i)} = \boldsymbol{B}_{k+1} \Delta \boldsymbol{x}^{(i)}$, $0 \leq i \leq k$.
- Notice that this set of equations is similar to the previous set of equations for $\boldsymbol{H}_{k+1}$, the only difference being that the roles of $\Delta \boldsymbol{x}^{(i)}$ and $\Delta \boldsymbol{g}^{(i)}$ are interchanged.
- Given any update formula for $\boldsymbol{H}_k$, a corresponding update formula for $\boldsymbol{B}_k$ can be found by interchanging the roles of $\boldsymbol{B}_k$ and $\boldsymbol{H}_k$ and of $\Delta \boldsymbol{g}^{(k)}$ and $\Delta \boldsymbol{x}^{(k)}$. In particular, the BFGS update for $\boldsymbol{B}_k$ corresponds to the DFP update for $\boldsymbol{H}_k$. Formulas related in this way are said to be ***dual*** or ***complementary***.

The BFGS Algorithm

- Recall that the DFP update for the approximation $\boldsymbol{H}_k$ of the inverse Hessian is

$$\boldsymbol{H}_{k+1}^{DFP} = \boldsymbol{H}_k + \frac{\Delta\boldsymbol{x}^{(k)}\Delta\boldsymbol{x}^{(k)T}}{\Delta\boldsymbol{x}^{(k)T}\Delta\boldsymbol{g}^{(k)}} - \frac{[\boldsymbol{H}_k\Delta\boldsymbol{g}^{(k)}][\boldsymbol{H}_k\Delta\boldsymbol{g}^{(k)}]^T}{\Delta\boldsymbol{g}^{(k)T}\boldsymbol{H}_k\Delta\boldsymbol{g}^{(k)}}$$

- Using the complementarity concept, we can easily obtain an update equation for the approximation $\boldsymbol{B}_k$ of the Hessian

$$\boldsymbol{B}_{k+1} = \boldsymbol{B}_k + \frac{\Delta\boldsymbol{g}^{(k)}\Delta\boldsymbol{g}^{(k)T}}{\Delta\boldsymbol{g}^{(k)T}\Delta\boldsymbol{x}^{(k)}} - \frac{[\boldsymbol{B}_k\Delta\boldsymbol{x}^{(k)}][\boldsymbol{B}_k\Delta\boldsymbol{x}^{(k)}]^T}{\Delta\boldsymbol{x}^{(k)T}\boldsymbol{B}_k\Delta\boldsymbol{x}^{(k)}}$$

- To obtain the BFGS update for the approximation of the inverse Hessian, we take the inverse of $\boldsymbol{B}_{k+1}$ to obtain

$$\begin{aligned}\boldsymbol{H}_{k+1}^{BFGS} &= (\boldsymbol{B}_{k+1})^{-1} \\ &= \left(\boldsymbol{B}_k + \frac{\Delta\boldsymbol{g}^{(k)}\Delta\boldsymbol{g}^{(k)T}}{\Delta\boldsymbol{g}^{(k)T}\Delta\boldsymbol{x}^{(k)}} - \frac{[\boldsymbol{B}_k\Delta\boldsymbol{x}^{(k)}][\boldsymbol{B}_k\Delta\boldsymbol{x}^{(k)}]^T}{\Delta\boldsymbol{x}^{(k)T}\boldsymbol{B}_k\Delta\boldsymbol{x}^{(k)}}\right)^{-1}\end{aligned}$$

The BFGS Algorithm

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

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

- From Lemma 11.1 it follows that if $\boldsymbol{A}^{-1}$ is known, then the inverse of the matrix $\boldsymbol{A}$ augmented by a rank one matrix can be obtained by a modification of the matrix $\boldsymbol{A}^{-1}$.

The BFGS Algorithm

- Applying Lemma 11.1 twice to $\boldsymbol{B}_{k+1}$ yields

$$\boldsymbol{H}_{k+1}^{BFGS} = \boldsymbol{H}_k + \left(1 + \frac{\Delta \boldsymbol{g}^{(k)T} \boldsymbol{H}_k \Delta \boldsymbol{g}^{(k)}}{\Delta \boldsymbol{g}^{(k)T} \Delta \boldsymbol{x}^{(k)}}\right) \frac{\Delta \boldsymbol{x}^{(k)} \Delta \boldsymbol{x}^{(k)T}}{\Delta \boldsymbol{x}^{(k)T} \Delta \boldsymbol{g}^{(k)}}$$
$$- \frac{\boldsymbol{H}_k \Delta \boldsymbol{g}^{(k)} \Delta \boldsymbol{x}^{(k)T} + (\boldsymbol{H}_k \Delta \boldsymbol{g}^{(k)} \Delta \boldsymbol{x}^{(k)T})^T}{\Delta \boldsymbol{g}^{(k)T} \Delta \boldsymbol{x}^{(k)}}$$

- Recall that for the quadratic case the DFP algorithm satisfies $\boldsymbol{H}_{k+1}^{DFP} \Delta \boldsymbol{g}^{(i)} = \boldsymbol{x}^{(i)}, 0 \le i \le k$. Therefore, the BFGS update for $\boldsymbol{B}_k$ satisfies $\boldsymbol{B}_{k+1} \Delta \boldsymbol{x}^{(i)} = \boldsymbol{g}^{(i)}, 0 \le i \le k$. By construction of the BFGS formula for $\boldsymbol{H}_{k+1}^{BFGS}$, we conclude that $\boldsymbol{H}_{k+1}^{BFGS} \Delta \boldsymbol{g}^{(i)} = \Delta \boldsymbol{x}^{(i)}, 0 \le i \le k$ 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 $\boldsymbol{g}^{(k)} \ne \mathbf{0}$ and $\boldsymbol{H}_k > 0$, then $\boldsymbol{H}_{k+1}^{BFGS} > 0$

Example

- The BFGS formula is often far more efficient than the DFP formula.
- Use the BFGS method to minimize $f(\boldsymbol{x}) = \frac{1}{2}\boldsymbol{x}^T\boldsymbol{Q}\boldsymbol{x} - \boldsymbol{x}^T\boldsymbol{b} + \log(\pi)$

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

- Take $\boldsymbol{H}_0 = \boldsymbol{I}_2$ and $\boldsymbol{x}_0 = [0, 0]^T$. Verify that $\boldsymbol{H}_2 = \boldsymbol{Q}^{-1}$.
- We have $$\boldsymbol{d}^{(0)} = -\boldsymbol{g}^{(0)} = -(\boldsymbol{Q}\boldsymbol{x}^{(0)} - \boldsymbol{b}) = \boldsymbol{b} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

  The objective function is a quadratic, and hence we can use the following formula to compute α_0

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

Example

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

  To compute $\boldsymbol{H}_1 = \boldsymbol{H}_1^{BFGS}$, we need the following quantities:

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

$$\boldsymbol{g}^{(1)} = \boldsymbol{Q}\boldsymbol{x}^{(1)} - \boldsymbol{b} = \begin{bmatrix} -3/2 \\ 0 \end{bmatrix}$$

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

  Therefore,

$$\boldsymbol{H}_1 = \boldsymbol{H}_0 + \left(1 + \frac{\Delta \boldsymbol{g}^{(0)T} \boldsymbol{H}_0 \Delta \boldsymbol{g}^{(0)}}{\Delta \boldsymbol{g}^{(0)T} \Delta \boldsymbol{x}^{(0)}}\right) \frac{\Delta \boldsymbol{x}^{(0)} \Delta \boldsymbol{x}^{(0)T}}{\Delta \boldsymbol{x}^{(0)T} \Delta \boldsymbol{g}^{(0)}} - \frac{\Delta \boldsymbol{x}^{(0)} \Delta \boldsymbol{g}^{(0)T} \boldsymbol{H}_0 + \boldsymbol{H}_0 \Delta \boldsymbol{g}^{(0)} \Delta \boldsymbol{x}^{(0)T}}{\Delta \boldsymbol{g}^{(0)T} \Delta \boldsymbol{x}^{(0)}} = \begin{bmatrix} 1 & 3/2 \\ 3/2 & 11/4 \end{bmatrix}$$

Example

- Hence, we have $\boldsymbol{d}^{(1)} = -\boldsymbol{H}_1 \boldsymbol{g}^{(1)} = \begin{bmatrix} 3/2 \\ 9/4 \end{bmatrix}$

  Therefore,

$$\alpha_1 = -\frac{\boldsymbol{g}^{(1)T}\boldsymbol{d}^{(1)}}{\boldsymbol{d}^{(1)T}\boldsymbol{Q}\boldsymbol{d}^{(1)}} = 2$$

$$\boldsymbol{x}^{(2)} = \boldsymbol{x}^{(1)} + \alpha_1 \boldsymbol{d}^{(1)} = \begin{bmatrix} 3 \\ 5 \end{bmatrix}$$

- Because our objective function is a quadratic on R^2, $\boldsymbol{x}^{(2)}$ is the minimizer. Notice that the gradient at $\boldsymbol{x}^{(2)}$ is $\mathbf{0}$; that is, $\boldsymbol{g}^{(2)} = \mathbf{0}$

Example

- To verify that $\boldsymbol{H}_2 = \boldsymbol{Q}^{-1}$, we compute

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

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

$$\boldsymbol{H}_2 = \boldsymbol{H}_1 + \left(1 + \frac{\Delta \boldsymbol{g}^{(1)T} \boldsymbol{H}_1 \Delta \boldsymbol{g}^{(1)}}{\Delta \boldsymbol{g}^{(1)T} \Delta \boldsymbol{x}^{(1)}}\right) \frac{\Delta \boldsymbol{x}^{(1)} \Delta \boldsymbol{x}^{(1)T}}{\Delta \boldsymbol{x}^{(1)T} \Delta \boldsymbol{g}^{(1)}} - \frac{\Delta \boldsymbol{x}^{(1)} \Delta \boldsymbol{g}^{(1)T} \boldsymbol{H}_1 + \boldsymbol{H}_1 \Delta \boldsymbol{g}^{(1)} \Delta \boldsymbol{x}^{(1)T}}{\Delta \boldsymbol{g}^{(1)T} \Delta \boldsymbol{x}^{(1)}} = \begin{bmatrix} 2 & 3 \\ 3 & 5 \end{bmatrix}$$

$$\Rightarrow \quad \boldsymbol{H}_2 \boldsymbol{Q} = \boldsymbol{Q} \boldsymbol{H}_2 = \boldsymbol{I}_2 \quad \Rightarrow \quad \boldsymbol{H}_2 = \boldsymbol{Q}^{-1}$$