最佳化方法與應用
MA5037-*

Chapter 6. Quasi-Newton Methods

Introduction

In the mid 1950s, W.C. Davidon, a physicist working at Argonne National Laboratory, was using the coordinate descent method (see Section 9.3) to perform a long optimization calculation. At that time computers were not very stable, and to Davidon's frustration, the computer system would always crash before the calculation was finished. So Davidon decided to find a way of accelerating the iteration. The algorithm he developed – the first quasi-Newton algorithm – turned out to be one of the most creative ideas in nonlinear optimization. It was soon demonstrated by Fletcher and Powell that the new algorithm was much faster and more reliable than the other existing methods, and this dramatic advance transformed nonlinear optimization overnight.

Introduction

During the following twenty years, numerous variants were proposed and hundreds of papers were devoted to their study. An interesting historical irony is that Davidon's paper [87] was not accepted for publication; it remained as a technical report for more than thirty years until it appeared in the first issue of the SIAM Journal on Optimization in 1991 [88].

Quasi-Newton methods, like steepest descent, require only the gradient of the objective function to be supplied at each iterate. By measuring the changes in gradients, they construct a model of the objective function that is good enough to produce superlinear convergence. The improvement over steepest descent is dramatic, especially on difficult problems.

Introduction

During the following twenty years, numerous variants were proposed and hundreds of papers were devoted to their study. An interesting historical irony is that Davidon's paper [87] was not accepted for publication; it remained as a technical report for more than thirty years until it appeared in the first issue of the SIAM Journal on Optimization in 1991 [88].

Quasi-Newton methods, like steepest descent, require only the gradient of the objective function to be supplied at each iterate. By measuring the changes in gradients, they construct a model of the objective function that is good enough to produce superlinear convergence. The improvement over steepest descent is dramatic, especially on difficult problems.

Introduction

Moreover, since second derivatives are not required, quasi-Newton methods are sometimes more efficient than Newton's method. Today, optimization software libraries contain a variety of quasi-Newton algorithms for solving unconstrained, constrained, and large-scale optimization problems. In this chapter we discuss quasi-Newton methods for small and medium-sized problems, and in Chapter 7 we consider their extension to the large-scale setting.

The development of automatic differentiation techniques has made it possible to use Newton's method without requiring users to supply second derivatives; see Chapter 8. Still, automatic differentiation tools may not be applicable in many situations, and it may be much more costly to work with second derivatives in automatic differentiation software than with the gradient. For these reasons, quasi-Newton methods remain appealing.

Introduction

Moreover, since second derivatives are not required, quasi-Newton methods are sometimes more efficient than Newton's method. Today, optimization software libraries contain a variety of quasi-Newton algorithms for solving unconstrained, constrained, and large-scale optimization problems. In this chapter we discuss quasi-Newton methods for small and medium-sized problems, and in Chapter 7 we consider their extension to the large-scale setting.

The development of automatic differentiation techniques has made it possible to use Newton's method without requiring users to supply second derivatives; see Chapter 8. Still, automatic differentiation tools may not be applicable in many situations, and it may be much more costly to work with second derivatives in automatic differentiation software than with the gradient. For these reasons, quasi-Newton methods remain appealing.

§6.1 The BFGS Method

The most popular quasi-Newton algorithm is the BFGS method, named for its discoverers Broyden, Fletcher, Goldfarb, and Shanno. In this section we derive this algorithm (and its close relative, the DFP algorithm) and describe its theoretical properties and practical implementation.

We begin the derivation by forming the following quadratic model of the objective function at the current iterate x_k:

$$m_k(p) = f_k + \nabla f_k^{\mathrm{T}} p + \frac{1}{2} p^{\mathrm{T}} B_k p \,. \tag{1}$$

Here B_k is an $n \times n$ symmetric positive definite matrix that will be revised or updated at every iteration. Note that the function value and gradient of this model at $p = 0$ match f_k and ∇f_k, respectively.

§6.1 The BFGS Method

The most popular quasi-Newton algorithm is the BFGS method, named for its discoverers Broyden, Fletcher, Goldfarb, and Shanno. In this section we derive this algorithm (and its close relative, the DFP algorithm) and describe its theoretical properties and practical implementation.

We begin the derivation by forming the following quadratic model of the objective function at the current iterate x_k:

$$m_k(p) = f_k + \nabla f_k^{\mathrm{T}} p + \frac{1}{2} p^{\mathrm{T}} B_k p \,. \tag{1}$$

Here B_k is an $n \times n$ symmetric positive definite matrix that will be revised or updated at every iteration. Note that the function value and gradient of this model at $p = 0$ match f_k and ∇f_k, respectively.

§6.1 The BFGS Method

The minimizer p_k of this convex quadratic model, which we can write explicitly as

$$p_k = -B_k^{-1}\nabla f_k\,, \tag{2}$$

is used as the search direction, and the new iterate is

$$x_{k+1} = x_k + \alpha_k p_k\,, \tag{3}$$

where the step length α_k is chosen to satisfy the Wolfe conditions. This iteration is quite similar to the line search Newton method; the key difference is that the approximate Hessian B_k is used in place of the true Hessian.

§6.1 The BFGS Method

Instead of computing B_k afresh at every iteration, Davidon proposed to update it in a simple manner to account for the **curvature measured** during the most recent step. Suppose that we have generated a new iterate x_{k+1} and wish to construct a new quadratic model, of the form

$$m_{k+1}(p) = f_{k+1} + \nabla f_{k+1}^{\mathrm{T}} p + \frac{1}{2} p^{\mathrm{T}} B_{k+1}\, p\,.$$

What requirements should we impose on B_{k+1}, based on the knowledge gained during the latest step? One reasonable requirement is that the gradient of m_{k+1} should match the gradient of the objective function f at the latest two iterates x_k and x_{k+1}. Since $\nabla m_{k+1}(0)$ is precisely ∇f_{k+1}, the second of these conditions is satisfied automatically. The first condition can be written mathematically as

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

§6.1 The BFGS Method

Instead of computing B_k afresh at every iteration, Davidon proposed to update it in a simple manner to account for the **curvature measured** during the most recent step. Suppose that we have generated a new iterate x_{k+1} and wish to construct a new quadratic model, of the form

$$m_{k+1}(p) = f_{k+1} + \nabla f_{k+1}^{\mathrm{T}} p + \frac{1}{2} p^{\mathrm{T}} B_{k+1}\, p\,.$$

What requirements should we impose on B_{k+1}, based on the knowledge gained during the latest step? One reasonable requirement is that the gradient of m_{k+1} should match the gradient of the objective function f at the latest two iterates x_k and x_{k+1}. Since $\nabla m_{k+1}(0)$ is precisely ∇f_{k+1}, the second of these conditions is satisfied automatically. The first condition can be written mathematically as

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

§6.1 The BFGS Method

By rearranging, we obtain

$$B_{k+1}\alpha_k p_k = \nabla f_{k+1} - \nabla f_k\,. \tag{4}$$

To simplify the notation it is useful to define the vectors

$$s_k = x_{k+1} - x_k = \alpha_k p_k\,, \quad y_k = \nabla f_{k+1} - \nabla f_k\,, \tag{5}$$

so that (4) becomes

$$B_{k+1} s_k = y_k\,. \tag{6}$$

We refer to this formula as the **secant** equation.

Given the displacement s_k and the change of gradients y_k, the secant equation requires that the symmetric positive definite matrix B_{k+1} map s_k into y_k. This will be possible **only if** s_k and y_k satisfy the curvature condition

$$s_k^{\mathrm{T}} y_k > 0\,, \tag{7}$$

as is easily seen by premultiplying (6) by s_k^{T}.

§6.1 The BFGS Method

By rearranging, we obtain

$$B_{k+1}\alpha_k p_k = \nabla f_{k+1} - \nabla f_k\,. \tag{4}$$

To simplify the notation it is useful to define the vectors

$$s_k = x_{k+1} - x_k = \alpha_k p_k\,, \quad y_k = \nabla f_{k+1} - \nabla f_k\,, \tag{5}$$

so that (4) becomes

$$B_{k+1}s_k = y_k\,. \tag{6}$$

We refer to this formula as the **secant** equation.

Given the displacement s_k and the change of gradients y_k, the secant equation requires that the symmetric positive definite matrix B_{k+1} map s_k into y_k. This will be possible **only if** s_k and y_k satisfy the curvature condition

$$s_k^{\mathrm{T}} y_k > 0\,, \tag{7}$$

as is easily seen by premultiplying (6) by s_k^{T}.

§6.1 The BFGS Method

When f is strongly convex, the inequality

$$s_k^{\mathrm{T}} y_k > 0 \tag{7}$$

will be satisfied for any two points x_k and x_{k+1}. However, this condition will not always hold for non-convex functions, and in this case we need to enforce (7) explicitly, by imposing restrictions on the line search procedure that chooses the step length α. In fact, the condition (7) is guaranteed to hold if we impose the Wolfe conditions or strong Wolfe conditions on the line search. To verify this claim, we note from (5) and the curvature condition that $\nabla f_{k+1}^{\mathrm{T}} s_k \geqslant c_2 \nabla f_k^{\mathrm{T}} s_k$, and therefore

$$y_k^{\mathrm{T}} s_k \geqslant (c_2 - 1)\alpha_k \nabla f_k^{\mathrm{T}} p_k \,. \tag{8}$$

Since $c_2 < 1$ and since p_k is a descent direction, the term on the right is positive, and the curvature condition (7) holds.

§6.1 The BFGS Method

When f is strongly convex, the inequality

$$s_k^{\mathrm{T}} y_k > 0 \tag{7}$$

will be satisfied for any two points x_k and x_{k+1}. However, this condition will not always hold for non-convex functions, and in this case we need to enforce (7) explicitly, by imposing restrictions on the line search procedure that chooses the step length α. In fact, the condition (7) is guaranteed to hold if we impose the Wolfe conditions or strong Wolfe conditions on the line search. To verify this claim, we note from (5) and the curvature condition that $\nabla f_{k+1}^{\mathrm{T}} s_k \geqslant c_2 \nabla f_k^{\mathrm{T}} s_k$, and therefore

$$y_k^{\mathrm{T}} s_k \geqslant (c_2 - 1)\alpha_k \nabla f_k^{\mathrm{T}} p_k \,. \tag{8}$$

Since $c_2 < 1$ and since p_k is a descent direction, the term on the right is positive, and the curvature condition (7) holds.

§6.1 The BFGS Method

When f is strongly convex, the inequality

$$s_k^{\mathrm{T}} y_k > 0 \tag{7}$$

will be satisfied for any two points x_k and x_{k+1}. However, this condition will not always hold for non-convex functions, and in this case we need to enforce (7) explicitly, by imposing restrictions on the line search procedure that chooses the step length α. In fact, the condition (7) is guaranteed to hold if we impose the Wolfe conditions or strong Wolfe conditions on the line search. To verify this claim, we note from (5) and the curvature condition that $\nabla f_{k+1}^{\mathrm{T}} s_k \geqslant c_2 \nabla f_k^{\mathrm{T}} s_k$, and therefore

$$y_k^{\mathrm{T}} s_k \geqslant (c_2 - 1)\alpha_k \nabla f_k^{\mathrm{T}} p_k\,. \tag{8}$$

Since $c_2 < 1$ and since p_k is a descent direction, the term on the right is positive, and the curvature condition (7) holds.

§6.1 The BFGS Method

When f is strongly convex, the inequality

$$s_k^{\mathrm{T}} y_k > 0 \tag{7}$$

will be satisfied for any two points x_k and x_{k+1}. However, this condition will not always hold for non-convex functions, and in this case we need to enforce (7) explicitly, by imposing restrictions on the line search procedure that chooses the step length α. In fact, the condition (7) is guaranteed to hold if we impose the Wolfe conditions or strong Wolfe conditions on the line search. To verify this claim, we note from (5) and the curvature condition that $\nabla f_{k+1}^{\mathrm{T}} s_k \geqslant c_2 \nabla f_k^{\mathrm{T}} s_k$, and therefore

$$y_k^{\mathrm{T}} s_k \geqslant (c_2 - 1)\alpha_k \nabla f_k^{\mathrm{T}} p_k \,. \tag{8}$$

Since $c_2 < 1$ and since p_k is a descent direction, the term on the right is positive, and the curvature condition (7) holds.

§6.1 The BFGS Method

When the curvature condition is satisfied, the secant equation (6) always has a solution B_{k+1}. In fact, it admits an infinite number of solutions, since the $\dfrac{n(n+1)}{2}$ degrees of freedom in a symmetric positive definite matrix exceed the n conditions imposed by the secant equation. The requirement of positive definiteness imposes n additional inequalities – all principal minors must be positive – but these conditions do not absorb the remaining degrees of freedom.

§6.1 The BFGS Method

To determine B_{k+1} uniquely, we impose the additional condition that among all symmetric matrices satisfying the secant equation, B_{k+1} is, in some sense, closest to the current matrix B_k. In other words, we solve the problem

$$\min_{B} \|B - B_k\| \qquad \text{subject to} \quad B = B^{\mathrm{T}} \text{ and } Bs_k = y_k\,, \tag{9}$$

where s_k and y_k satisfy

$$s_k^{\mathrm{T}} y_k > 0 \tag{7}$$

and B_k is symmetric and positive definite.

§6.1 The BFGS Method

Different matrix norms can be used in (9), and each norm gives rise to a different quasi-Newton method. A norm that allows easy solution of the minimization problem (9) and gives rise to a scale-invariant optimization method is the weighted Frobenius norm

$$\|A\|_W \equiv \|W^{1/2} A W^{1/2}\|_F \,, \tag{10}$$

where $\|\cdot\|_F$ is defined by $\|C\|_F^2 = \mathrm{tr}(C^{\mathrm{T}} C) = \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij}^2$. The weight matrix W can be chosen as any positive definite matrix satisfying $W y_k = s_k$. For concreteness, the reader can assume that

$$W = \bar{G}_k^{-1} \,,$$

where $\bar{G}_k$ is the average Hessian defined by

$$\bar{G}_k = \int_0^1 (\nabla^2 f)(x_k + \tau \alpha_k p_k)\, d\tau \,. \tag{11}$$

§6.1 The BFGS Method

Different matrix norms can be used in (9), and each norm gives rise to a different quasi-Newton method. A norm that allows easy solution of the minimization problem (9) and gives rise to a scale-invariant optimization method is the weighted Frobenius norm

$$\|A\|_W \equiv \|W^{1/2}AW^{1/2}\|_F, \tag{10}$$

where $\|\cdot\|_F$ is defined by $\|C\|_F^2 = \mathrm{tr}(C^{\mathrm{T}}C) = \sum_{i=1}^{n}\sum_{j=1}^{n} c_{ij}^2$. The weight matrix W can be chosen as any positive definite matrix satisfying $Wy_k = s_k$. For concreteness, the reader can assume that

$$W = \bar{G}_k^{-1},$$

where $\bar{G}_k$ is the average Hessian defined by

$$\bar{G}_k = \int_0^1 (\nabla^2 f)(x_k + \tau\alpha_k p_k)\, d\tau\,. \tag{11}$$

§6.1 The BFGS Method

Different matrix norms can be used in (9), and each norm gives rise to a different quasi-Newton method. A norm that allows easy solution of the minimization problem (9) and gives rise to a scale-invariant optimization method is the weighted Frobenius norm

$$\|A\|_W \equiv \|W^{1/2} A W^{1/2}\|_F , \tag{10}$$

where $\|\cdot\|_F$ is defined by $\|C\|_F^2 = \mathrm{tr}(C^{\mathrm{T}} C) = \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij}^2$. The weight matrix W can be chosen as any positive definite matrix satisfying $Wy_k = s_k$. For concreteness, the reader can assume that

$$W = \bar{G}_k^{-1} ,$$

where $\bar{G}_k$ is the average Hessian defined by

$$\bar{G}_k = \int_0^1 (\nabla^2 f)(x_k + \tau\alpha_k p_k)\, d\tau . \tag{11}$$

§6.1 The BFGS Method

The property

$$y_k = \bar{G}_k \alpha_k p_k = \bar{G}_k s_k \tag{12}$$

follows from Taylor's theorem. With this choice of weighting matrix W, the norm (10) is non-dimensional, which is a desirable property, since we do not wish the solution of (9) to depend on the units of the problem. With a weighting matrix W satisfying $Wy_k = s_k$ and this weighted norm, the unique solution of (9) is

$$\text{(DFP)} \quad B_{k+1} = (\mathrm{I} - \rho_k y_k s_k^{\mathrm{T}}) B_k (\mathrm{I} - \rho_k s_k y_k^{\mathrm{T}}) + \rho_k y_k y_k^{\mathrm{T}}, \tag{13}$$

with

$$\rho_k = \frac{1}{y_k^{\mathrm{T}} s_k}. \tag{14}$$

This formula is called **the DFP updating formula**, since it is the one originally proposed by **D**avidon in 1959, and subsequently studied, implemented, and popularized by **F**letcher and **P**owell.

§6.1 The BFGS Method

The property

$$y_k = \bar{G}_k \alpha_k p_k = \bar{G}_k s_k \tag{12}$$

follows from Taylor's theorem. With this choice of weighting matrix W, the norm (10) is non-dimensional, which is a desirable property, since we do not wish the solution of (9) to depend on the units of the problem. With a weighting matrix W satisfying $W y_k = s_k$ and this weighted norm, the unique solution of (9) is

$$\text{(DFP)} \quad B_{k+1} = (\mathrm{I} - \rho_k y_k s_k^{\mathrm{T}}) B_k (\mathrm{I} - \rho_k s_k y_k^{\mathrm{T}}) + \rho_k y_k y_k^{\mathrm{T}}, \tag{13}$$

with

$$\rho_k = \frac{1}{y_k^{\mathrm{T}} s_k}. \tag{14}$$

This formula is called **the DFP updating formula**, since it is the one originally proposed by **D**avidon in 1959, and subsequently studied, implemented, and popularized by **F**letcher and **P**owell.

§6.1 The BFGS Method

The inverse of B_k, which we denote by

$$H_k = B_k^{-1} \,,$$

is useful in the implementation of the method, since it allows the search direction (2) to be calculated by means of a simple matrix-vector multiplication. Using **the Sherman-Morrison-Woodbury formula**, we can derive the following expression for the update of the inverse Hessian approximation H_k that corresponds to the DFP update of B_k in (13):

$$\text{(DFP)} \qquad H_{k+1} = H_k - \frac{H_k y_k y_k^{\mathrm{T}} H_k}{y_k^{\mathrm{T}} H_k y_k} + \frac{s_k s_k^{\mathrm{T}}}{y_k^{\mathrm{T}} s_k} \,. \qquad (15)$$

§6.1 The BFGS Method

The inverse of B_k, which we denote by

$$H_k = B_k^{-1},$$

is useful in the implementation of the method, since it allows the search direction (2) to be calculated by means of a simple matrix-vector multiplication. Using **the Sherman-Morrison-Woodbury formula**, we can derive the following expression for the update of the inverse Hessian approximation H_k that corresponds to the DFP update of B_k in (13):

$$\text{(DFP)} \qquad H_{k+1} = H_k - \frac{H_k y_k y_k^{\mathrm{T}} H_k}{y_k^{\mathrm{T}} H_k y_k} + \frac{s_k s_k^{\mathrm{T}}}{y_k^{\mathrm{T}} s_k}. \tag{15}$$

§6.1 The BFGS Method

$$\text{(DFP)} \qquad H_{k+1} = H_k - \frac{H_k y_k y_k^{\mathrm{T}} H_k}{y_k^{\mathrm{T}} H_k y_k} + \frac{s_k s_k^{\mathrm{T}}}{y_k^{\mathrm{T}} s_k}\,. \tag{15}$$

Note that the last two terms in the right-hand-side of (15) are rank-one matrices, so that H_k undergoes a rank-two modification. It is easy to see that

$$\text{(DFP)} \qquad B_{k+1} = (\mathrm{I}-\rho_k y_k s_k^{\mathrm{T}}) B_k (\mathrm{I}-\rho_k s_k y_k^{\mathrm{T}}) + \rho_k y_k y_k^{\mathrm{T}}, \tag{13}$$

is also a rank-two modification of B_k. This is the fundamental idea of quasi-Newton updating: Instead of recomputing the approximate Hessians (or inverse Hessians) from scratch at every iteration, we apply a simple modification that combines the most recently observed information about the objective function with the existing knowledge embedded in our current Hessian approximation.

§6.1 The BFGS Method

$$\text{(DFP)} \qquad H_{k+1} = H_k - \frac{H_k y_k y_k^{\mathrm{T}} H_k}{y_k^{\mathrm{T}} H_k y_k} + \frac{s_k s_k^{\mathrm{T}}}{y_k^{\mathrm{T}} s_k} \,. \tag{15}$$

Note that the last two terms in the right-hand-side of (15) are rank-one matrices, so that H_k undergoes a rank-two modification. It is easy to see that

$$\text{(DFP)} \qquad B_{k+1} = (\mathrm{I} - \rho_k y_k s_k^{\mathrm{T}}) B_k (\mathrm{I} - \rho_k s_k y_k^{\mathrm{T}}) + \rho_k y_k y_k^{\mathrm{T}} , \tag{13}$$

is also a rank-two modification of B_k. This is the fundamental idea of quasi-Newton updating: Instead of recomputing the approximate Hessians (or inverse Hessians) from scratch at every iteration, we apply a simple modification that combines the most recently observed information about the objective function with the existing knowledge embedded in our current Hessian approximation.

§6.1 The BFGS Method

$$\text{(DFP)} \qquad H_{k+1} = H_k - \frac{H_k y_k y_k^{\mathrm{T}} H_k}{y_k^{\mathrm{T}} H_k y_k} + \frac{s_k s_k^{\mathrm{T}}}{y_k^{\mathrm{T}} s_k} \,. \tag{15}$$

Note that the last two terms in the right-hand-side of (15) are rank-one matrices, so that H_k undergoes a rank-two modification. It is easy to see that

$$\text{(DFP)} \qquad B_{k+1} = (\mathrm{I} - \rho_k y_k s_k^{\mathrm{T}}) B_k (\mathrm{I} - \rho_k s_k y_k^{\mathrm{T}}) + \rho_k y_k y_k^{\mathrm{T}} , \tag{13}$$

is also a rank-two modification of B_k. This is the fundamental idea of quasi-Newton updating: Instead of recomputing the approximate Hessians (or inverse Hessians) from scratch at every iteration, we apply a simple modification that combines the most recently observed information about the objective function with the existing knowledge embedded in our current Hessian approximation.

§6.1 The BFGS Method

- **The derivation of the DFP updating formula**

Let $\widetilde{s}_k = W^{-1/2}s_k$ and $\widetilde{y}_k = W^{1/2}y_k$, we find that $Wy_k = s_k$ if and only if $\widetilde{y}_k = \widetilde{s}_k$. Moreover, the condition $Bs_k = y_k$ becomes $W^{1/2}BW^{1/2}\widetilde{s}_k = \widetilde{y}_k$. For a given square matrix M, define $\widetilde{M} = W^{1/2}MW^{1/2}$. Then Problem (9) can be reformulated as

$$\min_{\widetilde{B}} \|\widetilde{B} - \widetilde{B}_k\|_F \qquad \text{subject to} \quad \widetilde{B} = \widetilde{B}^{\mathrm{T}} \text{ and } \widetilde{B}\widetilde{y}_k = \widetilde{y}_k\,.$$

Therefore, we look for a symmetric positive definiteness matrix $\widetilde{B}_{k+1}$ satisfying $(\mathrm{I} - \widetilde{B}_{k+1})\widetilde{y}_k = 0$ and minimizing the function

$$f(\widetilde{B}) \equiv \|\widetilde{B} - \widetilde{B}_k\|_F^2 = \mathrm{tr}\big((\widetilde{B} - \widetilde{B}_k)^{\mathrm{T}}(\widetilde{B} - \widetilde{B}_k)\big)\,.$$

We differentiate the function and find that $\widetilde{B}_{k+1}$ satisfies that

$$\mathrm{tr}\big((\widetilde{B}_{k+1} - \widetilde{B}_k)^{\mathrm{T}}\delta\widetilde{B}\big) = 0$$

whenever $\delta\widetilde{B}$ is symmetric and satisfies that $\delta\widetilde{B}\widetilde{y}_k = 0$.

§6.1 The BFGS Method

- **The derivation of the DFP updating formula**

Let $\widetilde{s}_k = W^{-1/2}s_k$ and $\widetilde{y}_k = W^{1/2}y_k$, we find that $Wy_k = s_k$ if and only if $\widetilde{y}_k = \widetilde{s}_k$. Moreover, the condition $Bs_k = y_k$ becomes $W^{1/2}BW^{1/2}\widetilde{s}_k = \widetilde{y}_k$. For a given square matrix M, define $\widetilde{M} = W^{1/2}MW^{1/2}$. Then Problem (9) can be reformulated as

$$\min_{\widetilde{B}} \|\widetilde{B} - \widetilde{B}_k\|_F \qquad \text{subject to} \quad \widetilde{B} = \widetilde{B}^{\mathrm{T}} \text{ and } \widetilde{B}\widetilde{y}_k = \widetilde{y}_k\,.$$

Therefore, we look for a symmetric positive definiteness matrix $\widetilde{B}_{k+1}$ satisfying $(\mathrm{I} - \widetilde{B}_{k+1})\widetilde{y}_k = 0$ and minimizing the function

$$f(\widetilde{B}) \equiv \|\widetilde{B} - \widetilde{B}_k\|_F^2 = \mathrm{tr}\big((\widetilde{B} - \widetilde{B}_k)^{\mathrm{T}}(\widetilde{B} - \widetilde{B}_k)\big)\,.$$

We differentiate the function and find that $\widetilde{B}_{k+1}$ satisfies that

$$\mathrm{tr}\big((\widetilde{B}_{k+1} - \widetilde{B}_k)^{\mathrm{T}}\delta\widetilde{B}\big) = 0$$

whenever $\delta\widetilde{B}$ is symmetric and satisfies that $\delta\widetilde{B}\,\widetilde{y}_k = 0$.

§6.1 The BFGS Method

- **The derivation of the DFP updating formula**

Let $\widetilde{s}_k = W^{-1/2}s_k$ and $\widetilde{y}_k = W^{1/2}y_k$, we find that $Wy_k = s_k$ if and only if $\widetilde{y}_k = \widetilde{s}_k$. Moreover, the condition $Bs_k = y_k$ becomes $W^{1/2}BW^{1/2}\widetilde{s}_k = \widetilde{y}_k$. For a given square matrix M, define $\widetilde{M} = W^{1/2}MW^{1/2}$. Then Problem (9) can be reformulated as

$$\min_{\widetilde{B}} \|\widetilde{B} - \widetilde{B}_k\|_F \qquad \text{subject to} \quad \widetilde{B} = \widetilde{B}^{\mathrm{T}} \text{ and } \widetilde{B}\,\widetilde{y}_k = \widetilde{y}_k\,.$$

Therefore, we look for a symmetric positive definiteness matrix $\widetilde{B}_{k+1}$ satisfying $(\mathrm{I} - \widetilde{B}_{k+1})\widetilde{y}_k = 0$ and minimizing the function

$$f(\widetilde{B}) \equiv \|\widetilde{B} - \widetilde{B}_k\|_F^2 = \mathrm{tr}\big((\widetilde{B} - \widetilde{B}_k)^{\mathrm{T}}(\widetilde{B} - \widetilde{B}_k)\big)\,.$$

We differentiate the function and find that $\widetilde{B}_{k+1}$ satisfies that

$$\mathrm{tr}\big((\widetilde{B}_{k+1} - \widetilde{B}_k)^{\mathrm{T}}\delta\widetilde{B}\big) = 0$$

whenever $\delta\widetilde{B}$ is symmetric and satisfies that $\delta\widetilde{B}\,\widetilde{y}_k = 0$.

§6.1 The BFGS Method

- **The derivation of the DFP updating formula**

Let $\widetilde{s}_k = W^{-1/2}s_k$ and $\widetilde{y}_k = W^{1/2}y_k$, we find that $Wy_k = s_k$ if and only if $\widetilde{y}_k = \widetilde{s}_k$. Moreover, the condition $Bs_k = y_k$ becomes $W^{1/2}BW^{1/2}\widetilde{s}_k = \widetilde{y}_k$. For a given square matrix M, define $\widetilde{M} = W^{1/2}MW^{1/2}$. Then Problem (9) can be reformulated as

$$\min_{\widetilde{B}} \|\widetilde{B} - \widetilde{B}_k\|_F \qquad \text{subject to} \quad \widetilde{B} = \widetilde{B}^{\mathrm{T}} \text{ and } \widetilde{B}\,\widetilde{y}_k = \widetilde{y}_k\,.$$

Therefore, we look for a symmetric positive definiteness matrix $\widetilde{B}_{k+1}$ satisfying $(\mathrm{I} - \widetilde{B}_{k+1})\widetilde{y}_k = 0$ and minimizing the function

$$f(\widetilde{B}) \equiv \|\widetilde{B} - \widetilde{B}_k\|_F^2 = \mathrm{tr}\big((\widetilde{B} - \widetilde{B}_k)^{\mathrm{T}}(\widetilde{B} - \widetilde{B}_k)\big)\,.$$

We differentiate the function and find that $\widetilde{B}_{k+1}$ satisfies that

$$\mathrm{tr}\big((\widetilde{B}_{k+1} - \widetilde{B}_k)^{\mathrm{T}}\delta\widetilde{B}\big) = 0$$

whenever $\delta\widetilde{B}$ is symmetric and satisfies that $\delta\widetilde{B}\,\widetilde{y}_k = 0$.

§6.1 The BFGS Method

Choose an orthogonal matrix O such that $\mathrm{O}\tilde{y}_k = \|\tilde{y}_k\|\mathrm{e}_n$, where $\mathrm{e}_n = [0, 0, \cdots, 0, 1]^{\mathrm{T}}$. By the fact that $\mathrm{tr}(\mathrm{O}M\mathrm{O}^{\mathrm{T}}) = \mathrm{tr}(M)$ for all M and $\mathrm{O}\,\delta\widetilde{B}\,\mathrm{O}^{\mathrm{T}}\mathrm{e}_n = 0$, we find that $\widetilde{B}$ satisfies

$$0 = \mathrm{tr}\big((\widetilde{B}_{k+1} - \widetilde{B}_k)^{\mathrm{T}}\delta\widetilde{B}\big) = \mathrm{tr}\big((\mathrm{O}(\widetilde{B}_{k+1} - \widetilde{B}_k)\mathrm{O}^{\mathrm{T}})^{\mathrm{T}}(\mathrm{O}\,\delta\widetilde{B}\,\mathrm{O}^{\mathrm{T}})\big)$$

whenever $\delta\widetilde{B}$ satisfies that the last row and the last column of $\mathrm{O}\,\delta\widetilde{B}\,\mathrm{O}^{\mathrm{T}}$ are zero. This implies that

$$\mathrm{O}(\widetilde{B}_{k+1} - \widetilde{B}_k)\mathrm{O}^{\mathrm{T}} = \begin{bmatrix} 0 & \cdots & 0 & a_{1n} \\ \vdots & \ddots & \vdots & \vdots \\ 0 & \cdots & 0 & a_{(n-1)n} \\ a_{1n} & \cdots & a_{(n-1)n} & a_{nn} \end{bmatrix}.$$

This shows that the minimizer $B_{k+1} (= W^{-1/2}\widetilde{B}_{k+1}W^{-1/2})$ is a rank-two modification of B_k.

§6.1 The BFGS Method

Choose an orthogonal matrix O such that $\mathrm{O}\tilde{y}_k = \|\tilde{y}_k\|\mathrm{e}_n$, where $\mathrm{e}_n = [0, 0, \cdots, 0, 1]^{\mathrm{T}}$. By the fact that $\mathrm{tr}(\mathrm{O}M\mathrm{O}^{\mathrm{T}}) = \mathrm{tr}(M)$ for all M and $\mathrm{O}\,\delta\widetilde{B}\,\mathrm{O}^{\mathrm{T}}\mathrm{e}_n = 0$, we find that $\widetilde{B}$ satisfies

$$0 = \mathrm{tr}\Big((\widetilde{B}_{k+1} - \widetilde{B}_k)^{\mathrm{T}}\delta\widetilde{B}\Big) = \mathrm{tr}\Big((\mathrm{O}(\widetilde{B}_{k+1} - \widetilde{B}_k)\mathrm{O}^{\mathrm{T}})^{\mathrm{T}}(\mathrm{O}\,\delta\widetilde{B}\,\mathrm{O}^{\mathrm{T}})\Big)$$

whenever $\delta\widetilde{B}$ satisfies that the last row and the last column of $\mathrm{O}\,\delta\widetilde{B}\,\mathrm{O}^{\mathrm{T}}$ are zero. This implies that

$$\mathrm{O}(\widetilde{B}_{k+1} - \widetilde{B}_k)\mathrm{O}^{\mathrm{T}} = \begin{bmatrix} 0 & \cdots & 0 & a_{1n} \\ \vdots & \ddots & \vdots & \vdots \\ 0 & \cdots & 0 & a_{(n-1)n} \\ a_{1n} & \cdots & a_{(n-1)n} & a_{nn} \end{bmatrix}.$$

This shows that the minimizer $B_{k+1} (= W^{-1/2}\widetilde{B}_{k+1}W^{-1/2})$ is a rank-two modification of B_k.

§6.1 The BFGS Method

For a given $n \times n$ matrix M, let $[M]_{(n-1)\times(n-1)}$ denote the $(n-1) \times (n-1)$ matrix obtained by deleting the last row and last column of M. Then the identity in the previous slide shows that

$$\big[\mathrm{O}\widetilde{B}_{k+1}\mathrm{O}^{\mathrm{T}}\big]_{(n-1)\times(n-1)} = \big[\mathrm{O}\widetilde{B}_{k}\mathrm{O}^{\mathrm{T}}\big]_{(n-1)\times(n-1)}\,.$$

To determine the last row and the last column of $\mathrm{O}\widetilde{B}_{k+1}\mathrm{O}^{\mathrm{T}}$, we note that the condition $\widetilde{B}_{k+1}\widetilde{y}_k = \widetilde{y}_k$ is equivalent to that

$$\mathrm{O}\widetilde{B}_{k+1}\mathrm{O}^{\mathrm{T}}\mathrm{e}_n = \mathrm{e}_n\,.$$

Therefore, the last row and last column of $\mathrm{O}\widetilde{B}_{k+1}\mathrm{O}^{\mathrm{T}}$ is e_n. This shows that

$$\mathrm{O}\widetilde{B}_{k+1}\mathrm{O}^{\mathrm{T}} = \begin{bmatrix} \big[\mathrm{O}\widetilde{B}_{k}\mathrm{O}^{\mathrm{T}}\big]_{(n-1)\times(n-1)} & 0 \\ 0 & 1 \end{bmatrix}. \tag{16}$$

§6.1 The BFGS Method

For a given $n \times n$ matrix M, let $[M]_{(n-1)\times(n-1)}$ denote the $(n-1)\times(n-1)$ matrix obtained by deleting the last row and last column of M. Then the identity in the previous slide shows that

$$\big[\mathrm{O}\widetilde{B}_{k+1}\mathrm{O}^{\mathrm{T}}\big]_{(n-1)\times(n-1)} = \big[\mathrm{O}\widetilde{B}_k\mathrm{O}^{\mathrm{T}}\big]_{(n-1)\times(n-1)}\,.$$

To determine the last row and the last column of $\mathrm{O}\widetilde{B}_{k+1}\mathrm{O}^{\mathrm{T}}$, we note that the condition $\widetilde{B}_{k+1}\widetilde{y}_k = \widetilde{y}_k$ is equivalent to that

$$\mathrm{O}\widetilde{B}_{k+1}\mathrm{O}^{\mathrm{T}}\mathrm{e}_n = \mathrm{e}_n\,.$$

Therefore, the last row and last column of $\mathrm{O}\widetilde{B}_{k+1}\mathrm{O}^{\mathrm{T}}$ is e_n. This shows that

$$\mathrm{O}\widetilde{B}_{k+1}\mathrm{O}^{\mathrm{T}} = \begin{bmatrix} \big[\mathrm{O}\widetilde{B}_k\mathrm{O}^{\mathrm{T}}\big]_{(n-1)\times(n-1)} & 0 \\ 0 & 1 \end{bmatrix}. \tag{16}$$

§6.1 The BFGS Method

For a given $n \times n$ matrix M, let $[M]_{(n-1)\times(n-1)}$ denote the $(n-1) \times (n-1)$ matrix obtained by deleting the last row and last column of M. Then the identity in the previous slide shows that

$$\left[\mathrm{O}\widetilde{B}_{k+1}\mathrm{O}^{\mathrm{T}}\right]_{(n-1)\times(n-1)} = \left[\mathrm{O}\widetilde{B}_k\mathrm{O}^{\mathrm{T}}\right]_{(n-1)\times(n-1)}.$$

To determine the last row and the last column of $\mathrm{O}\widetilde{B}_{k+1}\mathrm{O}^{\mathrm{T}}$, we note that the condition $\widetilde{B}_{k+1}\widetilde{y}_k = \widetilde{y}_k$ is equivalent to that

$$\mathrm{O}\widetilde{B}_{k+1}\mathrm{O}^{\mathrm{T}}\mathrm{e}_n = \mathrm{e}_n\,.$$

Therefore, the last row and last column of $\mathrm{O}\widetilde{B}_{k+1}\mathrm{O}^{\mathrm{T}}$ is e_n. This shows that

$$\mathrm{O}\widetilde{B}_{k+1}\mathrm{O}^{\mathrm{T}} = \begin{bmatrix} \left[\mathrm{O}\widetilde{B}_k\mathrm{O}^{\mathrm{T}}\right]_{(n-1)\times(n-1)} & 0 \\ 0 & 1 \end{bmatrix}. \tag{16}$$

§6.1 The BFGS Method

Note that

$$\text{(DFP)} \quad B_{k+1} = (\mathrm{I} - \rho_k y_k s_k^{\mathrm{T}}) B_k (\mathrm{I} - \rho_k s_k y_k^{\mathrm{T}}) + \rho_k y_k y_k^{\mathrm{T}}, \tag{13}$$

if and only if

$$\widetilde{B}_{k+1} = (\mathrm{I} - \rho_k \widetilde{y}_k \widetilde{s}_k^{\mathrm{T}}) \widetilde{B}_k (\mathrm{I} - \rho_k \widetilde{s}_k \widetilde{y}_k^{\mathrm{T}}) + \rho_k \widetilde{y}_k \widetilde{y}_k^{\mathrm{T}}.$$

Since $\widetilde{y}_k = \widetilde{s}_k$, it holds the identity

$$\rho_k = \frac{1}{y_k^{\mathrm{T}} s_k} = \frac{1}{\widetilde{y}_k^{\mathrm{T}} \widetilde{s}_k} = \frac{1}{\widetilde{y}_k^{\mathrm{T}} \widetilde{y}_k} = \|\widetilde{y}_k\|^{-2},$$

so to establish (13) it suffices to show that

$$\widetilde{B}_{k+1} = (\mathrm{I} - \bar{y}_k \bar{y}_k^{\mathrm{T}}) \widetilde{B}_k (\mathrm{I} - \bar{y}_k \bar{y}_k^{\mathrm{T}}) + \bar{y}_k \bar{y}_k^{\mathrm{T}}. \tag{13'}$$

where $\bar{y}_k = \widetilde{y}_k / \|\widetilde{y}_k\|$.

§6.1 The BFGS Method

Note that

$$\text{(DFP)} \quad B_{k+1} = (\mathrm{I} - \rho_k y_k s_k^{\mathrm{T}}) B_k (\mathrm{I} - \rho_k s_k y_k^{\mathrm{T}}) + \rho_k y_k y_k^{\mathrm{T}}, \tag{13}$$

if and only if

$$\widetilde{B}_{k+1} = (\mathrm{I} - \rho_k \widetilde{y}_k \widetilde{s}_k^{\mathrm{T}}) \widetilde{B}_k (\mathrm{I} - \rho_k \widetilde{s}_k \widetilde{y}_k^{\mathrm{T}}) + \rho_k \widetilde{y}_k \widetilde{y}_k^{\mathrm{T}}.$$

Since $\widetilde{y}_k = \widetilde{s}_k$, it holds the identity

$$\rho_k = \frac{1}{y_k^{\mathrm{T}} s_k} = \frac{1}{\widetilde{y}_k^{\mathrm{T}} \widetilde{s}_k} = \frac{1}{\widetilde{y}_k^{\mathrm{T}} \widetilde{y}_k} = \|\widetilde{y}_k\|^{-2},$$

so to establish (13) it suffices to show that

$$\widetilde{B}_{k+1} = (\mathrm{I} - \bar{y}_k \bar{y}_k^{\mathrm{T}}) \widetilde{B}_k (\mathrm{I} - \bar{y}_k \bar{y}_k^{\mathrm{T}}) + \bar{y}_k \bar{y}_k^{\mathrm{T}}. \tag{13'}$$

where $\bar{y}_k = \widetilde{y}_k / \|\widetilde{y}_k\|$.

§6.1 The BFGS Method

Note that

$$\text{(DFP)} \quad B_{k+1} = (\mathrm{I} - \rho_k y_k s_k^{\mathrm{T}}) B_k (\mathrm{I} - \rho_k s_k y_k^{\mathrm{T}}) + \rho_k y_k y_k^{\mathrm{T}}, \tag{13}$$

if and only if

$$\widetilde{B}_{k+1} = (\mathrm{I} - \rho_k \widetilde{y}_k \widetilde{s}_k^{\mathrm{T}}) \widetilde{B}_k (\mathrm{I} - \rho_k \widetilde{s}_k \widetilde{y}_k^{\mathrm{T}}) + \rho_k \widetilde{y}_k \widetilde{y}_k^{\mathrm{T}}.$$

Since $\widetilde{y}_k = \widetilde{s}_k$, it holds the identity

$$\rho_k = \frac{1}{y_k^{\mathrm{T}} s_k} = \frac{1}{\widetilde{y}_k^{\mathrm{T}} \widetilde{s}_k} = \frac{1}{\widetilde{y}_k^{\mathrm{T}} \widetilde{y}_k} = \|\widetilde{y}_k\|^{-2},$$

so to establish (13) it suffices to show that

$$\mathrm{O}\widetilde{B}_{k+1}\mathrm{O}^{\mathrm{T}} = (\mathrm{I} - \mathrm{e}_n \mathrm{e}_n^{\mathrm{T}}) \mathrm{O}\widetilde{B}_k \mathrm{O}^{\mathrm{T}} (\mathrm{I} - \mathrm{e}_n \mathrm{e}_n^{\mathrm{T}}) + \mathrm{e}_n \mathrm{e}_n^{\mathrm{T}}. \tag{13'}$$

where we use $\mathrm{O}\bar{y}_k = \mathrm{e}_n$ to conclude the identity. We note that (13') is equivalent to (16); thus the DFP updating formula is established.

§6.1 The BFGS Method

Note that

$$(\mathrm{DFP}) \quad B_{k+1} = (\mathrm{I} - \rho_k y_k s_k^{\mathrm{T}}) B_k (\mathrm{I} - \rho_k s_k y_k^{\mathrm{T}}) + \rho_k y_k y_k^{\mathrm{T}}, \tag{13}$$

if and only if

$$\widetilde{B}_{k+1} = (\mathrm{I} - \rho_k \widetilde{y}_k \widetilde{s}_k^{\mathrm{T}}) \widetilde{B}_k (\mathrm{I} - \rho_k \widetilde{s}_k \widetilde{y}_k^{\mathrm{T}}) + \rho_k \widetilde{y}_k \widetilde{y}_k^{\mathrm{T}}.$$

Since $\widetilde{y}_k = \widetilde{s}_k$, it holds the identity

$$\rho_k = \frac{1}{y_k^{\mathrm{T}} s_k} = \frac{1}{\widetilde{y}_k^{\mathrm{T}} \widetilde{s}_k} = \frac{1}{\widetilde{y}_k^{\mathrm{T}} \widetilde{y}_k} = \|\widetilde{y}_k\|^{-2},$$

so to establish (13) it suffices to show that

$$\mathrm{O}\widetilde{B}_{k+1}\mathrm{O}^{\mathrm{T}} = (\mathrm{I} - \mathrm{e}_n \mathrm{e}_n^{\mathrm{T}}) \mathrm{O}\widetilde{B}_k \mathrm{O}^{\mathrm{T}} (\mathrm{I} - \mathrm{e}_n \mathrm{e}_n^{\mathrm{T}}) + \mathrm{e}_n \mathrm{e}_n^{\mathrm{T}}. \tag{13'}$$

where we use $\mathrm{O}\bar{y}_k = \mathrm{e}_n$ to conclude the identity. We note that (13') is equivalent to (16); thus the DFP updating formula is established.

Sherman-Morrison-Woodbury Formula

Theorem

Let A be an $n \times n$ non-singular matrix, and U and V be matrices in $\mathbb{R}^{n\times p}$ for some p between 1 and n. If $\widehat{A} = A + UV^{\mathrm{T}}$, then $\widehat{A}$ is non-singular if and only if $(\mathrm{I} + V^{\mathrm{T}}A^{-1}U)$ is non-singular, and in this case we have

$$\widehat{A}^{-1} = A^{-1} - A^{-1}U(\mathrm{I} + V^{\mathrm{T}}A^{-1}U)^{-1}V^{\mathrm{T}}A^{-1}. \tag{17}$$

In particular, if the square non-singular matrix A undergoes a rank-one update to become

$$\bar{A} = A + ab^{\mathrm{T}},$$

where $a, b \in \mathbb{R}^n$, then if $\bar{A}$ is non-singular, we have

$$\bar{A}^{-1} = A^{-1} - \frac{A^{-1}ab^{\mathrm{T}}A^{-1}}{1 + b^{\mathrm{T}}A^{-1}a}. \tag{18}$$

Sherman-Morrison-Woodbury Formula

Proof.

We write the linear system $(A+UV^{\mathrm{T}})x=d$ as

$$\begin{bmatrix} A & U \\ V^{\mathrm{T}} & -\mathrm{I}_{p\times p} \end{bmatrix} \begin{bmatrix} x \\ \xi \end{bmatrix} = \begin{bmatrix} d \\ 0 \end{bmatrix}$$

where $\xi = V^{\mathrm{T}}x$. Note that the $(n+p)\times(n+p)$ matrix above can be decomposed as

$$\begin{bmatrix} A & U \\ V^{\mathrm{T}} & -\mathrm{I}_{p\times p} \end{bmatrix} = \begin{bmatrix} \mathrm{I}_{n\times n} & 0_{n\times p} \\ V^{\mathrm{T}}A^{-1} & \mathrm{I}_{p\times p} \end{bmatrix} \begin{bmatrix} A & U \\ 0_{p\times n} & -(\mathrm{I}_{p\times p}+V^{\mathrm{T}}A^{-1}U) \end{bmatrix};$$

thus the linear system $(A+UV^{\mathrm{T}})x=d$ is uniquely solvable if and only if the linear system

$$\begin{bmatrix} A & U \\ 0_{p\times n} & -(\mathrm{I}_{p\times p}+V^{\mathrm{T}}A^{-1}U) \end{bmatrix} \begin{bmatrix} x \\ \xi \end{bmatrix} = \begin{bmatrix} d \\ -V^{\mathrm{T}}A^{-1}d \end{bmatrix}$$

is uniquely solvable. □

Sherman-Morrison-Woodbury Formula

Proof.

We write the linear system $(A+UV^{\mathrm{T}})x=d$ as

$$\begin{bmatrix} A & U \\ V^{\mathrm{T}} & -\mathrm{I}_{p\times p}\end{bmatrix}\begin{bmatrix} x \\ \xi \end{bmatrix} = \begin{bmatrix} d \\ 0 \end{bmatrix}$$

where $\xi = V^{\mathrm{T}}x$. Note that the $(n+p)\times(n+p)$ matrix above can be decomposed as

$$\begin{bmatrix} A & U \\ V^{\mathrm{T}} & -\mathrm{I}_{p\times p}\end{bmatrix} = \begin{bmatrix} \mathrm{I}_{n\times n} & 0_{n\times p} \\ V^{\mathrm{T}}A^{-1} & \mathrm{I}_{p\times p}\end{bmatrix}\begin{bmatrix} A & U \\ 0_{p\times n} & -(\mathrm{I}_{p\times p}+V^{\mathrm{T}}A^{-1}U)\end{bmatrix};$$

thus the linear system $(A+UV^{\mathrm{T}})x=d$ is uniquely solvable if and only if the linear system

$$\begin{bmatrix} A & U \\ 0_{p\times n} & -(\mathrm{I}_{p\times p}+V^{\mathrm{T}}A^{-1}U)\end{bmatrix}\begin{bmatrix} x \\ \xi \end{bmatrix} = \begin{bmatrix} d \\ -V^{\mathrm{T}}A^{-1}d \end{bmatrix}$$

is uniquely solvable. □

Sherman-Morrison-Woodbury Formula

Proof (cont'd).

Nevertheless, by the invertibility of A, the linear system

$$\begin{bmatrix} A & U \\ 0_{p\times n} & -(\mathrm{I}_{p\times p}+V^{\mathrm{T}}A^{-1}U) \end{bmatrix} \begin{bmatrix} x \\ \xi \end{bmatrix} = \begin{bmatrix} d \\ -V^{\mathrm{T}}A^{-1}d \end{bmatrix}$$

is uniquely solvable if and only if the system

$$(\mathrm{I}_{p\times p}+V^{\mathrm{T}}A^{-1}U)\xi = V^{\mathrm{T}}A^{-1}d$$

is uniquely solvable so we establish that $\widehat{A} = A + UV^{\mathrm{T}}$ is non-singular if and only if $(\mathrm{I} + V^{\mathrm{T}}A^{-1}U)$ is non-singular. In this case,

$$\xi = (\mathrm{I}_{p\times p}+V^{\mathrm{T}}A^{-1}U)^{-1}V^{\mathrm{T}}A^{-1}d;$$

thus, by solving $Ax = d - U\xi$, we obtain that the solution of the linear system $(A + UV^{\mathrm{T}})x = d$ is given by

$$x = A^{-1}\big[\mathrm{I} - U(\mathrm{I}_{p\times p}+V^{\mathrm{T}}A^{-1}U)^{-1}V^{\mathrm{T}}A^{-1}\big]d. \qquad \square$$

Sherman-Morrison-Woodbury Formula

Proof (cont'd).

Nevertheless, by the invertibility of A, the linear system

$$\begin{bmatrix} A & U \\ 0_{p\times n} & -(\mathrm{I}_{p\times p}+V^{\mathrm{T}}A^{-1}U) \end{bmatrix} \begin{bmatrix} x \\ \xi \end{bmatrix} = \begin{bmatrix} d \\ -V^{\mathrm{T}}A^{-1}d \end{bmatrix}$$

is uniquely solvable if and only if the system

$$(\mathrm{I}_{p\times p}+V^{\mathrm{T}}A^{-1}U)\xi = V^{\mathrm{T}}A^{-1}d$$

is uniquely solvable so we establish that $\widehat{A} = A + UV^{\mathrm{T}}$ is non-singular if and only if $(\mathrm{I} + V^{\mathrm{T}}A^{-1}U)$ is non-singular. In this case,

$$\xi = (\mathrm{I}_{p\times p}+V^{\mathrm{T}}A^{-1}U)^{-1}V^{\mathrm{T}}A^{-1}d;$$

thus, by solving $Ax = d - U\xi$, we obtain that the solution of the linear system $(A + UV^{\mathrm{T}})x = d$ is given by

$$x = A^{-1}\big[\mathrm{I} - U(\mathrm{I}_{p\times p}+V^{\mathrm{T}}A^{-1}U)^{-1}V^{\mathrm{T}}A^{-1}\big]d.$$

□

Sherman-Morrison-Woodbury Formula

Proof (cont'd).

Nevertheless, by the invertibility of A, the linear system

$$\begin{bmatrix} A & U \\ 0_{p\times n} & -(\mathrm{I}_{p\times p}+V^{\mathrm{T}}A^{-1}U) \end{bmatrix} \begin{bmatrix} x \\ \xi \end{bmatrix} = \begin{bmatrix} d \\ -V^{\mathrm{T}}A^{-1}d \end{bmatrix}$$

is uniquely solvable if and only if the system

$$(\mathrm{I}_{p\times p}+V^{\mathrm{T}}A^{-1}U)\xi = V^{\mathrm{T}}A^{-1}d$$

is uniquely solvable so we establish that $\widehat{A} = A + UV^{\mathrm{T}}$ is non-singular if and only if $(\mathrm{I} + V^{\mathrm{T}}A^{-1}U)$ is non-singular. In this case,

$$\xi = (\mathrm{I}_{p\times p}+V^{\mathrm{T}}A^{-1}U)^{-1}V^{\mathrm{T}}A^{-1}d;$$

thus, by solving $Ax = d - U\xi$, we obtain that the solution of the linear system $(A + UV^{\mathrm{T}})x = d$ is given by

$$x = A^{-1}\big[\mathrm{I} - U(\mathrm{I}_{p\times p}+V^{\mathrm{T}}A^{-1}U)^{-1}V^{\mathrm{T}}A^{-1}\big]d. \qquad \square$$

Sherman-Morrison-Woodbury Formula

We can use the Sherman-Morrison-Woodbury formula to solve linear systems of the form $\widehat{A}x = d$. Since

$$\begin{aligned} x &= A^{-1}\big[\mathrm{I} - U(\mathrm{I}_{p\times p} + V^{\mathrm{T}}A^{-1}U)^{-1}V^{\mathrm{T}}A^{-1}\big]d \\ &= A^{-1}d - (A^{-1}U)\big[\mathrm{I}_{p\times p} + V^{\mathrm{T}}(A^{-1}U)\big]^{-1}V^{\mathrm{T}}(A^{-1}d)\,, \end{aligned}$$

we see that x can be found by solving $(p+1)$ linear systems with the matrix A (to obtain $A^{-1}d$ and $A^{-1}U$), inverting the $p\times p$ matrix $\mathrm{I} + V^{\mathrm{T}}A^{-1}U$, and performing some elementary matrix algebra. Inversion of the $p \times p$ matrix $\mathrm{I} + V^{\mathrm{T}}A^{-1}U$ is inexpensive when $p \ll n$.

§6.1 The BFGS Method

- **The derivation of the DFP updating formula for H_k**

We expand the DFP updating formula for B_k

$$\text{(DFP)} \quad B_{k+1} = (\mathrm{I} - \rho_k y_k s_k^{\mathrm{T}}) B_k (\mathrm{I} - \rho_k s_k y_k^{\mathrm{T}}) + \rho_k y_k y_k^{\mathrm{T}}, \tag{13}$$

as

$$\begin{aligned} B_{k+1} &= B_k - \rho_k y_k s_k^{\mathrm{T}} B_k - \rho_k B_k s_k y_k^{\mathrm{T}} + \rho_k^2 y_k (s_k^{\mathrm{T}} B_k s_k) y_k^{\mathrm{T}} + \rho_k y_k y_k^{\mathrm{T}} \\ &= B_k - \rho_k y_k (B_k s_k)^{\mathrm{T}} - \rho_k (B_k s_k) y_k^{\mathrm{T}} + \rho_k (1 + \rho_k s_k^{\mathrm{T}} B_k s_k) y_k y_k^{\mathrm{T}} \\ &= B_k - \rho_k y_k (B_k s_k)^{\mathrm{T}} + \rho_k (\mu_k y_k - B_k s_k) y_k^{\mathrm{T}} \\ &= B_k + \Big[-\rho_k y_k \,\vdots\, \rho_k (\mu_k y_k - B_k s_k) \Big] \Big[B_k s_k \,\vdots\, y_k \Big]^{\mathrm{T}}, \end{aligned}$$

where $\mu_k = 1 + \rho_k s_k^{\mathrm{T}} B_k s_k$.

§6.1 The BFGS Method

Let $A = B_k$, $U = \left[-\rho_k y_k \vdots \rho_k(\mu_k y_k - B_k s_k)\right]$ and $V = \left[B_k s_k \vdots y_k\right]$. Then $B_{k+1} = A + UV^{\mathrm{T}}$. Since

$$A^{-1}U = \left[-\rho_k H_k y_k \vdots \rho_k(\mu_k H_k y_k - s_k)\right], \quad V^{\mathrm{T}}A^{-1} = \begin{bmatrix} s_k^{\mathrm{T}} \\ \cdots \\ y_k^{\mathrm{T}} H_k \end{bmatrix},$$

and

$$\mathrm{I} + V^{\mathrm{T}}A^{-1}U = \begin{bmatrix} 0 & 1 \\ -\rho_k y_k^{\mathrm{T}} H_k y_k & \rho_k \mu_k y_k^{\mathrm{T}} H_k y_k \end{bmatrix},$$

by the Sherman-Morrison-Woodbury formula we obtain that

$$H_{k+1} = H_k - \frac{\rho_k}{\rho_k y_k^{\mathrm{T}} H_k y_k}\left[-H_k y_k \vdots \mu_k H_k y_k - s_k\right]\begin{bmatrix} \rho_k \mu_k y_k^{\mathrm{T}} H_k y_k & -1 \\ \rho_k y_k^{\mathrm{T}} H_k y_k & 0 \end{bmatrix}\begin{bmatrix} s_k^{\mathrm{T}} \\ \cdots \\ y_k^{\mathrm{T}} H_k \end{bmatrix}$$

$$= H_k - \left[-H_k y_k \vdots \mu_k H_k y_k - s_k\right]\begin{bmatrix} \rho_k \mu_k & -\dfrac{1}{y_k^{\mathrm{T}} H_k y_k} \\ \rho_k & 0 \end{bmatrix}\begin{bmatrix} s_k^{\mathrm{T}} \\ \cdots \\ y_k^{\mathrm{T}} H_k \end{bmatrix}.$$

§6.1 The BFGS Method

Let $A = B_k$, $U = \big[-\rho_k y_k \vdots \rho_k(\mu_k y_k - B_k s_k)\big]$ and $V = \big[B_k s_k \vdots y_k\big]$. Then $B_{k+1} = A + UV^{\mathrm{T}}$. Since

$$A^{-1}U = \Big[-\rho_k H_k y_k \vdots \rho_k(\mu_k H_k y_k - s_k)\Big], \quad V^{\mathrm{T}}A^{-1} = \begin{bmatrix} s_k^{\mathrm{T}} \\ \cdots \\ y_k^{\mathrm{T}} H_k \end{bmatrix},$$

and

$$\mathrm{I} + V^{\mathrm{T}}A^{-1}U = \begin{bmatrix} 0 & 1 \\ -\rho_k y_k^{\mathrm{T}} H_k y_k & \rho_k \mu_k y_k^{\mathrm{T}} H_k y_k \end{bmatrix},$$

by the Sherman-Morrison-Woodbury formula we obtain that

$$H_{k+1} = H_k - \frac{\rho_k}{\rho_k y_k^{\mathrm{T}} H_k y_k}\Big[-H_k y_k \vdots \mu_k H_k y_k - s_k\Big]\begin{bmatrix} \rho_k \mu_k y_k^{\mathrm{T}} H_k y_k & -1 \\ \rho_k y_k^{\mathrm{T}} H_k y_k & 0 \end{bmatrix}\begin{bmatrix} s_k^{\mathrm{T}} \\ \cdots \\ y_k^{\mathrm{T}} H_k \end{bmatrix}$$

$$= H_k - \Big[-H_k y_k \vdots \mu_k H_k y_k - s_k\Big]\begin{bmatrix} \rho_k \mu_k & -\dfrac{1}{y_k^{\mathrm{T}} H_k y_k} \\ \rho_k & 0 \end{bmatrix}\begin{bmatrix} s_k^{\mathrm{T}} \\ \cdots \\ y_k^{\mathrm{T}} H_k \end{bmatrix}.$$

§6.1 The BFGS Method

Expanding the product of the matrices,

$$H_{k+1} = H_k - \left[-H_k y_k \vdots \mu_k H_k y_k - s_k \right] \begin{bmatrix} \rho_k \mu_k & -\dfrac{1}{y_k^{\mathrm{T}} H_k y_k} \\ \rho_k & 0 \end{bmatrix} \begin{bmatrix} s_k^{\mathrm{T}} \\ \cdots \\ y_k^{\mathrm{T}} H_k \end{bmatrix}$$

$$= H_k - \left[-H_k y_k \vdots \mu_k H_k y_k - s_k \right] \begin{bmatrix} \rho_k \mu_k s_k^{\mathrm{T}} - \dfrac{y_k^{\mathrm{T}} H_k}{y_k^{\mathrm{T}} H_k y_k} \\ \cdots\cdots\cdots\cdots\cdots \\ \rho_k s_k^{\mathrm{T}} \end{bmatrix}$$

$$= H_k + H_k y_k \Big(\rho_k \mu_k s_k^{\mathrm{T}} - \frac{y_k^{\mathrm{T}} H_k}{y_k^{\mathrm{T}} H_k y_k} \Big) - (\mu_k H_k y_k - s_k) \rho_k s_k^{\mathrm{T}}$$

$$= H_k + \rho_k \mu_k H_k y_k s_k^{\mathrm{T}} - \frac{H_k y_k y_k^{\mathrm{T}} H_k}{y_k^{\mathrm{T}} H_k y_k} - \rho_k \mu_k H_k y_k s_k^{\mathrm{T}} + \rho_k s_k s_k^{\mathrm{T}}$$

$$= H_k - \frac{H_k y_k y_k^{\mathrm{T}} H_k}{y_k^{\mathrm{T}} H_k y_k} + \rho_k s_k s_k^{\mathrm{T}},$$

which is exactly the DFP updating formula for H_k.

§6.1 The BFGS Method

Expanding the product of the matrices,

$$\begin{aligned}
H_{k+1} &= H_k - \left[-H_k y_k \,\vdots\, \mu_k H_k y_k - s_k \right] \begin{bmatrix} \rho_k \mu_k & -\dfrac{1}{y_k^{\mathrm{T}} H_k y_k} \\ \rho_k & 0 \end{bmatrix} \begin{bmatrix} s_k^{\mathrm{T}} \\ \cdots \\ y_k^{\mathrm{T}} H_k \end{bmatrix} \\
&= H_k - \left[-H_k y_k \,\vdots\, \mu_k H_k y_k - s_k \right] \begin{bmatrix} \rho_k \mu_k s_k^{\mathrm{T}} - \dfrac{y_k^{\mathrm{T}} H_k}{y_k^{\mathrm{T}} H_k y_k} \\ \cdots\cdots\cdots\cdots \\ \rho_k s_k^{\mathrm{T}} \end{bmatrix} \\
&= H_k + H_k y_k \Big(\rho_k \mu_k s_k^{\mathrm{T}} - \frac{y_k^{\mathrm{T}} H_k}{y_k^{\mathrm{T}} H_k y_k} \Big) - (\mu_k H_k y_k - s_k) \rho_k s_k^{\mathrm{T}} \\
&= H_k + \rho_k \mu_k H_k y_k s_k^{\mathrm{T}} - \frac{H_k y_k y_k^{\mathrm{T}} H_k}{y_k^{\mathrm{T}} H_k y_k} - \rho_k \mu_k H_k y_k s_k^{\mathrm{T}} + \rho_k s_k s_k^{\mathrm{T}} \\
&= H_k - \frac{H_k y_k y_k^{\mathrm{T}} H_k}{y_k^{\mathrm{T}} H_k y_k} + \rho_k s_k s_k^{\mathrm{T}} ,
\end{aligned}$$

which is exactly the DFP updating formula for H_k.

§6.1 The BFGS Method

The DFP updating formula is quite effective, but it was soon superseded by the BFGS formula, which is presently considered to be the most effective of all quasi-Newton updating formulae. BFGS updating can be derived by making a simple change in the argument that led to (13). Instead of imposing conditions on the Hessian approximations B_k, we impose similar conditions on their inverses H_k. The updated approximation H_{k+1} must be symmetric and positive definite, and must satisfy the secant equation (6), now written as

$$H_{k+1} y_k = s_k \,.$$

The condition of closeness to H_k is now specified by the following analogue of (9):

$$\min_{H} \|H - H_k\| \qquad \text{subject to} \quad H = H^{\mathrm{T}}, \, H y_k = s_k \,. \tag{19}$$

§6.1 The BFGS Method

The DFP updating formula is quite effective, but it was soon superseded by the BFGS formula, which is presently considered to be the most effective of all quasi-Newton updating formulae. BFGS updating can be derived by making a simple change in the argument that led to (13). Instead of imposing conditions on the Hessian approximations B_k, we impose similar conditions on their inverses H_k. The updated approximation H_{k+1} must be symmetric and positive definite, and must satisfy the secant equation (6), now written as

$$H_{k+1} y_k = s_k \,.$$

The condition of closeness to H_k is now specified by the following analogue of (9):

$$\min_{H} \|H - H_k\| \qquad \text{subject to} \quad H = H^{\mathrm{T}}, H y_k = s_k \,. \tag{19}$$

§6.1 The BFGS Method

The norm is again the weighted Frobenius norm described above, where the weight matrix W is now any matrix satisfying $Ws_k = y_k$. The unique solution H_{k+1} to (19) is given by

$$\text{(BFGS)} \qquad H_{k+1} = (\mathrm{I} - \rho_k s_k y_k^{\mathrm{T}}) H_k (I - \rho_k y_k s_k^{\mathrm{T}}) + \rho_k s_k s_k^{\mathrm{T}}\,, \qquad (20)$$

with ρ_k defined by (14).

How should we choose the initial approximation H_0? Unfortunately, there is no magic formula that works well in all cases. We can use specific information about the problem, for instance

1. H_0 is the inverse of an approximate Hessian at x_0;
2. H_0 is the identity matrix;
3. H_0 is a multiple of the identity matrix, where the multiple is chosen to reflect the scaling of the variables.

§6.1 The BFGS Method

The norm is again the weighted Frobenius norm described above, where the weight matrix W is now any matrix satisfying $Ws_k = y_k$. The unique solution H_{k+1} to (19) is given by

$$\text{(BFGS)} \qquad H_{k+1} = (\mathrm{I} - \rho_k s_k y_k^{\mathrm{T}}) H_k (I - \rho_k y_k s_k^{\mathrm{T}}) + \rho_k s_k s_k^{\mathrm{T}}, \qquad (20)$$

with ρ_k defined by (14).

How should we choose the initial approximation H_0? Unfortunately, there is no magic formula that works well in all cases. We can use specific information about the problem, for instance

1. H_0 is the inverse of an approximate Hessian at x_0;
2. H_0 is the identity matrix;
3. H_0 is a multiple of the identity matrix, where the multiple is chosen to reflect the scaling of the variables.

§6.1 The BFGS Method

Algorithm 6.1 (BFGS Method).

Given starting point x_0, convergence tolerance $\varepsilon > 0$, inverse Hessian approximation H_0;

$k \leftarrow 0$;

while $\|\nabla f_k\| > \varepsilon$;

Compute search direction

$$p_k = -H_k \nabla f_k\,; \tag{21}$$

Set $x_{k+1} = x_k + \alpha_k p_k$, where α_k is computed from a line search procedure to satisfy the Wolfe conditions;

Define $s_k = x_{k+1} - x_k$ and $y_k = \nabla f_{k+1} - \nabla f_k$;

Compute H_{k+1} by means of (20);

$k \leftarrow k + 1$;

end (while)

§6.1 The BFGS Method

Each iteration can be performed at a cost of $\mathcal{O}(n^2)$ arithmetic operations (plus the cost of function and gradient evaluations); there are no $\mathcal{O}(n^3)$ operations such as linear system solves or matrix-matrix operations. The algorithm is robust, and its rate of convergence is superlinear (whose proof will be given in Section 6.4), which is fast enough for most practical purposes. Even though Newton's method converges more rapidly (that is, quadratically), its cost per iteration usually is higher, because of its need for second derivatives and solution of a linear system.

§6.1 The BFGS Method

Each iteration can be performed at a cost of $\mathcal{O}(n^2)$ arithmetic operations (plus the cost of function and gradient evaluations); there are no $\mathcal{O}(n^3)$ operations such as linear system solves or matrix-matrix operations. The algorithm is robust, and its rate of convergence is superlinear (whose proof will be given in Section 6.4), which is fast enough for most practical purposes. Even though Newton's method converges more rapidly (that is, quadratically), its cost per iteration usually is higher, because of its need for second derivatives and solution of a linear system.

§6.1 The BFGS Method

We can derive a version of the BFGS algorithm that works with the Hessian approximation B_k rather than H_k. The update formula for B_k is obtained by simply applying the Sherman-Morrison-Woodbury formula to (20) to obtain

$$\text{(BFGS)} \qquad B_{k+1} = B_k - \frac{B_k s_k s_k^{\mathrm{T}} B_k}{s_k^{\mathrm{T}} B_k s_k} + \frac{y_k y_k^{\mathrm{T}}}{y_k^{\mathrm{T}} s_k}\,. \tag{22}$$

A naive implementation of this variant is not efficient for unconstrained minimization, because it requires the system $B_k p_k = -\nabla f_k$ to be solved for the step p_k, thereby increasing the cost of the step computation to $\mathcal{O}(n^3)$. We discuss later, however, that less expensive implementations of this variant are possible by updating Cholesky factors of B_k.

§6.1 The BFGS Method

We can derive a version of the BFGS algorithm that works with the Hessian approximation B_k rather than H_k. The update formula for B_k is obtained by simply applying the Sherman-Morrison-Woodbury formula to (20) to obtain

$$\text{(BFGS)} \qquad B_{k+1} = B_k - \frac{B_k s_k s_k^{\mathrm{T}} B_k}{s_k^{\mathrm{T}} B_k s_k} + \frac{y_k y_k^{\mathrm{T}}}{y_k^{\mathrm{T}} s_k}. \tag{22}$$

A naive implementation of this variant is not efficient for unconstrained minimization, because it requires the system $B_k p_k = -\nabla f_k$ to be solved for the step p_k, thereby increasing the cost of the step computation to $\mathcal{O}(n^3)$. We discuss later, however, that less expensive implementations of this variant are possible by updating Cholesky factors of B_k.

§6.1 The BFGS Method

- **Properties of the BFGS Method**

It is usually easy to observe the superlinear rate of convergence of the BFGS method on practical problems. Below, we report the last few iterations of the steepest descent, BFGS, and an inexact Newton method on Rosenbrock's function

$$f(x) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2 .$$

The table gives the value of $\|x_k - x_*\|$. The Wolfe conditions were imposed on the step length in all three methods. From the starting point $(-1.2, 1)$, the steepest descent method required 5264 iterations, whereas BFGS and Newton took only 34 and 21 iterations, respectively to reduce the gradient norm to 10^{-5}.

§6.1 The BFGS Method

steepest descent	BFGS	Newton
1.827e-04	1.70e-03	3.48e-02
1.826e-04	1.17e-03	1.44e-02
1.824e-04	1.34e-04	1.82e-04
1.823e-04	1.01e-06	1.17e-08

§6.1 The BFGS Method

Note that the minimization problem (19) that gives rise to the BFGS update formula does not explicitly require the updated Hessian approximation to be positive definite. Nevertheless, note that $y_k^{\mathrm{T}} s_k$ is positive, so that the updating formula

$$\text{(BFGS)} \qquad H_{k+1} = (\mathrm{I} - \rho_k s_k y_k^{\mathrm{T}}) H_k (I - \rho_k y_k s_k^{\mathrm{T}}) + \rho_k s_k s_k^{\mathrm{T}}, \tag{20}$$

is well-defined. For any nonzero vector z, we have

$$z^{\mathrm{T}} H_{k+1} z = w^{\mathrm{T}} H_k w + \rho_k (z^{\mathrm{T}} s_k)^2 \geqslant 0,$$

where we have defined $w = z - \rho_k y_k (s_k^{\mathrm{T}} z)$. The right hand side can be zero only if $s_k^{\mathrm{T}} z = 0$, but in this case $w = z \neq 0$, which implies that the first term is greater than zero. Therefore, we establish that H_{k+1} (obtained by the updating formula (20)) is positive definite whenever H_k is positive definite.

§6.1 The BFGS Method

Note that the minimization problem (19) that gives rise to the BFGS update formula does not explicitly require the updated Hessian approximation to be positive definite. Nevertheless, note that $y_k^{\mathrm{T}} s_k$ is positive, so that the updating formula

$$\text{(BFGS)} \qquad H_{k+1} = (\mathrm{I} - \rho_k s_k y_k^{\mathrm{T}}) H_k (I - \rho_k y_k s_k^{\mathrm{T}}) + \rho_k s_k s_k^{\mathrm{T}} \,, \tag{20}$$

is well-defined. For any nonzero vector z, we have

$$z^{\mathrm{T}} H_{k+1} z = w^{\mathrm{T}} H_k w + \rho_k (z^{\mathrm{T}} s_k)^2 \geqslant 0 \,,$$

where we have defined $w = z - \rho_k y_k (s_k^{\mathrm{T}} z)$. The right hand side can be zero only if $s_k^{\mathrm{T}} z = 0$, but in this case $w = z \neq 0$, which implies that the first term is greater than zero. Therefore, we establish that H_{k+1} (obtained by the updating formula (20)) is positive definite whenever H_k is positive definite.

§6.1 The BFGS Method

Note that the minimization problem (19) that gives rise to the BFGS update formula does not explicitly require the updated Hessian approximation to be positive definite. Nevertheless, note that $y_k^{\mathrm{T}} s_k$ is positive, so that the updating formula

$$\text{(BFGS)} \qquad H_{k+1} = (\mathrm{I} - \rho_k s_k y_k^{\mathrm{T}}) H_k (I - \rho_k y_k s_k^{\mathrm{T}}) + \rho_k s_k s_k^{\mathrm{T}} , \tag{20}$$

is well-defined. For any nonzero vector z, we have

$$z^{\mathrm{T}} H_{k+1} z = w^{\mathrm{T}} H_k w + \rho_k (z^{\mathrm{T}} s_k)^2 \geqslant 0 ,$$

where we have defined $w = z - \rho_k y_k (s_k^{\mathrm{T}} z)$. The right hand side can be zero only if $s_k^{\mathrm{T}} z = 0$, but in this case $w = z \neq 0$, which implies that the first term is greater than zero. Therefore, we establish that H_{k+1} (obtained by the updating formula (20)) is positive definite whenever H_k is positive definite.

§6.1 The BFGS Method

Note that the minimization problem (19) that gives rise to the BFGS update formula does not explicitly require the updated Hessian approximation to be positive definite. Nevertheless, note that $y_k^{\mathrm{T}} s_k$ is positive, so that the updating formula

$$\text{(BFGS)} \qquad H_{k+1} = (\mathrm{I} - \rho_k s_k y_k^{\mathrm{T}}) H_k (I - \rho_k y_k s_k^{\mathrm{T}}) + \rho_k s_k s_k^{\mathrm{T}}, \tag{20}$$

is well-defined. For any nonzero vector z, we have

$$z^{\mathrm{T}} H_{k+1} z = w^{\mathrm{T}} H_k w + \rho_k (z^{\mathrm{T}} s_k)^2 \geqslant 0,$$

where we have defined $w = z - \rho_k y_k (s_k^{\mathrm{T}} z)$. The right hand side can be zero only if $s_k^{\mathrm{T}} z = 0$, but in this case $w = z \neq 0$, which implies that the first term is greater than zero. Therefore, we establish that H_{k+1} (obtained by the updating formula (20)) is positive definite whenever H_k is positive definite.

§6.1 The BFGS Method

To make quasi-Newton updating formulae invariant to transformations in the variables (such as scaling transformations), it is necessary for the objectives (9) and (19) to be invariant under the same transformations. The choice of the weighting matrices W used to define the norms in (9) and (19) ensures that this condition holds. Many other choices of the weighting matrix W are possible, each one of them giving a different update formula. However, despite intensive searches, no formula has been found that is significantly more effective than BFGS.

The BFGS method has many interesting properties when applied to quadratic functions. We discuss these properties later in the more general context of the Broyden family of updating formulae, of which BFGS is a special case.

§6.1 The BFGS Method

To make quasi-Newton updating formulae invariant to transformations in the variables (such as scaling transformations), it is necessary for the objectives (9) and (19) to be invariant under the same transformations. The choice of the weighting matrices W used to define the norms in (9) and (19) ensures that this condition holds. Many other choices of the weighting matrix W are possible, each one of them giving a different update formula. However, despite intensive searches, no formula has been found that is significantly more effective than BFGS.

The BFGS method has many interesting properties when applied to quadratic functions. We discuss these properties later in the more general context of the Broyden family of updating formulae, of which BFGS is a special case.

§6.1 The BFGS Method

To make quasi-Newton updating formulae invariant to transformations in the variables (such as scaling transformations), it is necessary for the objectives (9) and (19) to be invariant under the same transformations. The choice of the weighting matrices W used to define the norms in (9) and (19) ensures that this condition holds. Many other choices of the weighting matrix W are possible, each one of them giving a different update formula. However, despite intensive searches, no formula has been found that is significantly more effective than BFGS.

The BFGS method has many interesting properties when applied to quadratic functions. We discuss these properties later in the more general context of the Broyden family of updating formulae, of which BFGS is a special case.

§6.1 The BFGS Method

It is reasonable to ask whether there are situations in which the updating formula such as

$$\text{(BFGS)} \qquad H_{k+1} = (\mathrm{I} - \rho_k s_k y_k^{\mathrm{T}}) H_k (I - \rho_k y_k s_k^{\mathrm{T}}) + \rho_k s_k s_k^{\mathrm{T}} , \tag{20}$$

can produce bad results. If at some iteration the matrix H_k becomes a poor approximation to the true inverse Hessian, is there any hope of correcting it? For example, when the inner product $y_k^{\mathrm{T}} s_k$ is tiny (but positive), then it follows from (20) that H_{k+1} contains very large elements. Is this behavior reasonable? A related question concerns the rounding errors that occur in finite-precision implementation of these methods. Can these errors grow to the point of erasing all useful information in the quasi-Newton approximate Hessian?

§6.1 The BFGS Method

These questions have been studied analytically and experimentally, and it is now known that the BFGS formula has very effective self-correcting properties. If the matrix H_k incorrectly estimates the curvature in the objective function, and if this bad estimate slows down the iteration, then the Hessian approximation will tend to correct itself within a few steps. It is also known that the DFP method is **less effective** in correcting bad Hessian approximations; this property is believed to be the reason for its poorer practical performance. The self-correcting properties of BFGS hold only when an adequate line search is performed. In particular, the Wolfe line search conditions ensure that the gradients are sampled at points that allow the model (1) to capture appropriate curvature information.

§6.1 The BFGS Method

These questions have been studied analytically and experimentally, and it is now known that the BFGS formula has very effective self-correcting properties. If the matrix H_k incorrectly estimates the curvature in the objective function, and if this bad estimate slows down the iteration, then the Hessian approximation will tend to correct itself within a few steps. It is also known that the DFP method is **less effective** in correcting bad Hessian approximations; this property is believed to be the reason for its poorer practical performance. The self-correcting properties of BFGS hold only when an adequate line search is performed. In particular, the Wolfe line search conditions ensure that the gradients are sampled at points that allow the model (1) to capture appropriate curvature information.

§6.1 The BFGS Method

These questions have been studied analytically and experimentally, and it is now known that the BFGS formula has very effective self-correcting properties. If the matrix H_k incorrectly estimates the curvature in the objective function, and if this bad estimate slows down the iteration, then the Hessian approximation will tend to correct itself within a few steps. It is also known that the DFP method is **less effective** in correcting bad Hessian approximations; this property is believed to be the reason for its poorer practical performance. The self-correcting properties of BFGS hold only when an adequate line search is performed. In particular, the Wolfe line search conditions ensure that the gradients are sampled at points that allow the model (1) to capture appropriate curvature information.

§6.1 The BFGS Method

It is interesting to note that the DFP and BFGS updating formulae are duals of each other, in the sense that one can be obtained from the other by the interchanges $s \leftrightarrow y$, $B \leftrightarrow H$. This symmetry is not surprising, given the manner in which we derived these methods above.

§6.1 The BFGS Method

- **Implementation**

A few details and enhancements need to be added to Algorithm 6.1 to produce an efficient implementation. The line search, which should satisfy either the Wolfe conditions or the strong Wolfe conditions, should always try the step length $\alpha_k = 1$ first, because this step length will eventually always be accepted (under certain conditions), thereby producing superlinear convergence of the overall algorithm. Computational observations strongly suggest that it is more economical, in terms of function evaluations, to perform a fairly inaccurate line search. The values $c_1 = 10^{-4}$ and $c_2 = 0.9$ are commonly used in the Wolfe condition.

§6.1 The BFGS Method

As mentioned earlier, the initial matrix H_0 often is set to some multiple $\beta \mathrm{I}$ of the identity, but there is no good general strategy for choosing the multiple β. If β is too large, so that the first step $p_0 = -\beta g_0$ is too long, many function evaluations may be required to find a suitable value for the step length α_0. Some software asks the user to prescribe a value δ for the norm of the first step, and then set $H_0 = \delta \|g_0\|^{-1} \mathrm{I}$ to achieve this norm.

A heuristic that is often quite effective is to scale the starting matrix after the first step has been computed but before the first BFGS update is performed. We change the provisional value $H_0 = \mathrm{I}$ by setting

$$H_0 \leftarrow \frac{y_k^{\mathrm{T}} s_k}{y_k^{\mathrm{T}} y_k} \mathrm{I}, \tag{23}$$

before applying the updating formula (20) to obtain H_1.

§6.1 The BFGS Method

As mentioned earlier, the initial matrix H_0 often is set to some multiple $\beta \mathrm{I}$ of the identity, but there is no good general strategy for choosing the multiple β. If β is too large, so that the first step $p_0 = -\beta g_0$ is too long, many function evaluations may be required to find a suitable value for the step length α_0. Some software asks the user to prescribe a value δ for the norm of the first step, and then set $H_0 = \delta \|g_0\|^{-1} \mathrm{I}$ to achieve this norm.

A heuristic that is often quite effective is to scale the starting matrix after the first step has been computed but before the first BFGS update is performed. We change the provisional value $H_0 = \mathrm{I}$ by setting

$$H_0 \leftarrow \frac{y_k^{\mathrm{T}} s_k}{y_k^{\mathrm{T}} y_k} \mathrm{I}, \tag{23}$$

before applying the updating formula (20) to obtain H_1.

§6.1 The BFGS Method

Formula (23) attempts to make the size of H_0 similar to that of $(\nabla^2 f)(x_0)^{-1}$, in the following sense. Assuming that the average Hessian defined in (11) is positive definite, there exists a square root $\bar{G}_k^{1/2}$ satisfying $\bar{G}_k = \bar{G}_k^{1/2}\bar{G}_k^{1/2}$. Therefore, by defining $z_k = \bar{G}_k^{1/2} s_k$ and using the relation $y_k = \bar{G}_k s_k$, we have

$$\frac{y_k^{\mathrm{T}} s_k}{y_k^{\mathrm{T}} y_k} = \frac{(\bar{G}_k^{1/2} s_k)^{\mathrm{T}} \bar{G}_k^{1/2} s_k}{(\bar{G}_k^{1/2} s_k)^{\mathrm{T}} \bar{G}_k \bar{G}_k^{1/2} s_k} = \frac{z_k^{\mathrm{T}} z_k}{z_k^{\mathrm{T}} \bar{G}_k z_k}. \tag{24}$$

The reciprocal of (24) is an approximation to one of the eigenvalues of $\bar{G}_k$, which in turn is close to an eigenvalue of $(\nabla^2 f)(x_k)$. Hence, the quotient (24) itself approximates an eigenvalue of $(\nabla^2 f)(x_k)^{-1}$. Other scaling factors can be used in (23), but the one presented here appears to be the most successful in practice.

§6.1 The BFGS Method

Formula (23) attempts to make the size of H_0 similar to that of $(\nabla^2 f)(x_0)^{-1}$, in the following sense. Assuming that the average Hessian defined in (11) is positive definite, there exists a square root $\bar{G}_k^{1/2}$ satisfying $\bar{G}_k = \bar{G}_k^{1/2}\bar{G}_k^{1/2}$. Therefore, by defining $z_k = \bar{G}_k^{1/2} s_k$ and using the relation $y_k = \bar{G}_k s_k$, we have

$$\frac{y_k^{\mathrm{T}} s_k}{y_k^{\mathrm{T}} y_k} = \frac{(\bar{G}_k^{1/2} s_k)^{\mathrm{T}} \bar{G}_k^{1/2} s_k}{(\bar{G}_k^{1/2} s_k)^{\mathrm{T}} \bar{G}_k \bar{G}_k^{1/2} s_k} = \frac{z_k^{\mathrm{T}} z_k}{z_k^{\mathrm{T}} \bar{G}_k z_k}. \tag{24}$$

The reciprocal of (24) is an approximation to one of the eigenvalues of $\bar{G}_k$, which in turn is close to an eigenvalue of $(\nabla^2 f)(x_k)$. Hence, the quotient (24) itself approximates an eigenvalue of $(\nabla^2 f)(x_k)^{-1}$. Other scaling factors can be used in (23), but the one presented here appears to be the most successful in practice.

§6.1 The BFGS Method

In (22) we gave an update formula

$$\text{(BFGS)} \qquad B_{k+1} = B_k - \frac{B_k s_k s_k^{\mathrm{T}} B_k}{s_k^{\mathrm{T}} B_k s_k} + \frac{y_k y_k^{\mathrm{T}}}{y_k^{\mathrm{T}} s_k}. \qquad (22)$$

for a BFGS method that works with the Hessian approximation B_k instead of the inverse Hessian approximation H_k. An efficient implementation of this approach does not store B_k explicitly, but rather the Cholesky factorization $L_k D_k L_k^{\mathrm{T}}$ of this matrix. A formula that updates the factors L_k and D_k directly in $\mathcal{O}(n^2)$ operations can be derived from (22). Since the linear system $B_k p_k = -\nabla f_k$ also can be solved in $\mathcal{O}(n^2)$ operations (by performing triangular substitutions with L_k and L_k^{T} and a diagonal substitution with D_k), the total cost is quite similar to the variant described in Algorithm 6.1.

§6.1 The BFGS Method

A potential advantage of this alternative strategy is that it gives us the option of modifying diagonal elements in the D_k factor if they are not sufficiently large, to prevent instability when we divide by these elements during the calculation of p_k. However, computational experience suggests **no real advantages** for this variant, and we prefer the simpler strategy of Algorithm 6.1.

§6.1 The BFGS Method

The performance of the BFGS method can degrade if the line search is not based on the Wolfe conditions. For example, some software implements an Armijo backtracking line search (see Section 3.1): The unit step length $\alpha_k = 1$ is tried first and is successively decreased until the sufficient decrease condition is satisfied. For this strategy, there is no guarantee that the curvature condition $y_k^{\mathrm{T}} s_k > 0$ (7) will be satisfied by the chosen step, since a step length greater than 1 may be required to satisfy this condition. To cope with this shortcoming, some implementations simply skip the BFGS update by setting $H_{k+1} = H_k$ when $y_k^{\mathrm{T}} s_k$ is negative or too close to zero. This approach is **not** recommended, because the updates may be skipped much too often to allow H_k to capture important curvature information for the objective function f.

§6.1 The BFGS Method

The performance of the BFGS method can degrade if the line search is not based on the Wolfe conditions. For example, some software implements an Armijo backtracking line search (see Section 3.1): The unit step length $\alpha_k = 1$ is tried first and is successively decreased until the sufficient decrease condition is satisfied. For this strategy, there is no guarantee that the curvature condition $y_k^{\mathrm{T}} s_k > 0$ (7) will be satisfied by the chosen step, since a step length greater than 1 may be required to satisfy this condition. To cope with this shortcoming, some implementations simply skip the BFGS update by setting $H_{k+1} = H_k$ when $y_k^{\mathrm{T}} s_k$ is negative or too close to zero. This approach is **not** recommended, because the updates may be skipped much too often to allow H_k to capture important curvature information for the objective function f.

§6.1 The BFGS Method

The performance of the BFGS method can degrade if the line search is not based on the Wolfe conditions. For example, some software implements an Armijo backtracking line search (see Section 3.1): The unit step length $\alpha_k = 1$ is tried first and is successively decreased until the sufficient decrease condition is satisfied. For this strategy, there is no guarantee that the curvature condition $y_k^{\mathrm{T}} s_k > 0$ (7) will be satisfied by the chosen step, since a step length greater than 1 may be required to satisfy this condition. To cope with this shortcoming, some implementations simply skip the BFGS update by setting $H_{k+1} = H_k$ when $y_k^{\mathrm{T}} s_k$ is negative or too close to zero. This approach is **not** recommended, because the updates may be skipped much too often to allow H_k to capture important curvature information for the objective function f.

§6.1 The BFGS Method

The performance of the BFGS method can degrade if the line search is not based on the Wolfe conditions. For example, some software implements an Armijo backtracking line search (see Section 3.1): The unit step length $\alpha_k = 1$ is tried first and is successively decreased until the sufficient decrease condition is satisfied. For this strategy, there is no guarantee that the curvature condition $y_k^{\mathrm{T}} s_k > 0$ (7) will be satisfied by the chosen step, since a step length greater than 1 may be required to satisfy this condition. To cope with this shortcoming, some implementations simply skip the BFGS update by setting $H_{k+1} = H_k$ when $y_k^{\mathrm{T}} s_k$ is negative or too close to zero. This approach is **not** recommended, because the updates may be skipped much too often to allow H_k to capture important curvature information for the objective function f.

§6.2 The SR1 Method

In the BFGS and DFP updating formulae, the updated matrix B_{k+1} (or H_{k+1}) differs from its predecessor B_k (or H_k) by a rank-2 matrix. In fact, as we now show, there is a simpler rank-1 update that maintains symmetry of the matrix and allows it to satisfy the secant equation. Unlike the rank-two update formulae, this symmetric-rank-1, or SR1, update does not guarantee that the updated matrix maintains positive definiteness. Good numerical results have been obtained with algorithms based on SR1, so we derive it here and investigate its properties.

§6.2 The SR1 Method

The symmetric rank-1 update has the general form $B_{k+1} = B_k + \sigma v v^{\mathrm{T}}$, where σ is either $+1$ or -1, and σ and v are chosen so that B_{k+1} satisfies the secant equation $y_k = B_{k+1} s_k$. By substituting into this equation, we obtain

$$y_k = B_k s_k + \left[\sigma v^{\mathrm{T}} s_k\right] v. \tag{25}$$

Since the term in brackets is a scalar, we deduce that v must be a multiple of $y_k - B_k s_k$; that is, $v = \delta(y_k - B_k s_k)$ for some scalar δ. By substituting this form of v into (25), we obtain

$$(y_k - B_k s_k) = \sigma\delta^2 \left[s_k^{\mathrm{T}}(y_k - B_k s_k)\right](y_k - B_k s_k), \tag{26}$$

and it is clear that this equation is satisfied if (and only if) we choose the parameters δ and σ to be

$$\sigma = \mathrm{sign}\left[s_k^{\mathrm{T}}(y_k - B_k s_k)\right], \quad \delta = \pm\left|s_k^{\mathrm{T}}(y_k - B_k s_k)\right|^{-1/2}.$$

§6.2 The SR1 Method

The symmetric rank-1 update has the general form $B_{k+1} = B_k + \sigma v v^{\mathrm{T}}$, where σ is either $+1$ or -1, and σ and v are chosen so that B_{k+1} satisfies the secant equation $y_k = B_{k+1} s_k$. By substituting into this equation, we obtain

$$y_k = B_k s_k + \left[\sigma v^{\mathrm{T}} s_k\right] v\,. \tag{25}$$

Since the term in brackets is a scalar, we deduce that v must be a multiple of $y_k - B_k s_k$; that is, $v = \delta(y_k - B_k s_k)$ for some scalar δ. By substituting this form of v into (25), we obtain

$$(y_k - B_k s_k) = \sigma\delta^2 \left[s_k^{\mathrm{T}}(y_k - B_k s_k)\right](y_k - B_k s_k)\,, \tag{26}$$

and it is clear that this equation is satisfied if (and only if) we choose the parameters δ and σ to be

$$\sigma = \operatorname{sign}\left[s_k^{\mathrm{T}}(y_k - B_k s_k)\right]\,, \quad \delta = \pm\left|s_k^{\mathrm{T}}(y_k - B_k s_k)\right|^{-1/2}.$$

§6.2 The SR1 Method

The symmetric rank-1 update has the general form $B_{k+1} = B_k + \sigma v v^{\mathrm{T}}$, where σ is either $+1$ or -1, and σ and v are chosen so that B_{k+1} satisfies the secant equation $y_k = B_{k+1} s_k$. By substituting into this equation, we obtain

$$y_k = B_k s_k + \left[\sigma v^{\mathrm{T}} s_k\right] v\,. \tag{25}$$

Since the term in brackets is a scalar, we deduce that v must be a multiple of $y_k - B_k s_k$; that is, $v = \delta(y_k - B_k s_k)$ for some scalar δ. By substituting this form of v into (25), we obtain

$$(y_k - B_k s_k) = \sigma\delta^2 \left[s_k^{\mathrm{T}}(y_k - B_k s_k)\right](y_k - B_k s_k)\,, \tag{26}$$

and it is clear that this equation is satisfied if (and only if) we choose the parameters δ and σ to be

$$\sigma = \operatorname{sign}\left[s_k^{\mathrm{T}}(y_k - B_k s_k)\right], \quad \delta = \pm\left|s_k^{\mathrm{T}}(y_k - B_k s_k)\right|^{-1/2}.$$

§6.2 The SR1 Method

The symmetric rank-1 update has the general form $B_{k+1} = B_k + \sigma v v^{\mathrm{T}}$, where σ is either $+1$ or -1, and σ and v are chosen so that B_{k+1} satisfies the secant equation $y_k = B_{k+1} s_k$. By substituting into this equation, we obtain

$$y_k = B_k s_k + \left[\sigma v^{\mathrm{T}} s_k\right] v\,. \tag{25}$$

Since the term in brackets is a scalar, we deduce that v must be a multiple of $y_k - B_k s_k$; that is, $v = \delta(y_k - B_k s_k)$ for some scalar δ. By substituting this form of v into (25), we obtain

$$(y_k - B_k s_k) = \sigma\delta^2 \left[s_k^{\mathrm{T}}(y_k - B_k s_k)\right](y_k - B_k s_k)\,, \tag{26}$$

and it is clear that this equation is satisfied if (and only if) we choose the parameters δ and σ to be

$$\sigma = \mathrm{sign}\left[s_k^{\mathrm{T}}(y_k - B_k s_k)\right], \quad \delta = \pm\left|s_k^{\mathrm{T}}(y_k - B_k s_k)\right|^{-1/2}.$$

§6.2 The SR1 Method

Hence, we have shown that the only symmetric rank-1 updating formula that satisfies the secant equation is given by

$$\text{(SR1)} \qquad B_{k+1} = B_k + \frac{(y_k - B_k s_k)(y_k - B_k s_k)^{\mathrm{T}}}{(y_k - B_k s_k)^{\mathrm{T}} s_k}\,. \tag{27}$$

By applying the Sherman-Morrison formula, we obtain the corresponding update formula for the inverse Hessian approximation H_k:

$$\text{(SR1)} \qquad H_{k+1} = H_k + \frac{(s_k - H_k y_k)(s_k - H_k y_k)^{\mathrm{T}}}{(s_k - H_k y_k)^{\mathrm{T}} y_k}\,. \tag{28}$$

This derivation is so simple that the SR1 formula has been rediscovered a number of times.

§6.2 The SR1 Method

It is easy to see that even if B_k is positive definite, B_{k+1} may not have the same property. (The same is, of course, true of H_k.) This observation was considered a major drawback in the early days of nonlinear optimization when only line search iterations were used. However, with the advent of **trust-region** methods, the SR1 updating formula has proved to be quite useful, and its ability to generate indefinite Hessian approximations can actually be regarded as one of its chief advantages.

The main drawback of SR1 updating is that the denominator in (27) or (28) can vanish. In fact, even when the objective function is a convex quadratic, there may be steps on which there is no symmetric rank-1 update that satisfies the secant equation. It pays to reexamine the derivation above in the light of this observation.

§6.2 The SR1 Method

It is easy to see that even if B_k is positive definite, B_{k+1} may not have the same property. (The same is, of course, true of H_k.) This observation was considered a major drawback in the early days of nonlinear optimization when only line search iterations were used. However, with the advent of **trust-region** methods, the SR1 updating formula has proved to be quite useful, and its ability to generate indefinite Hessian approximations can actually be regarded as one of its chief advantages.

The main drawback of SR1 updating is that the denominator in (27) or (28) can vanish. In fact, even when the objective function is a convex quadratic, there may be steps on which there is no symmetric rank-1 update that satisfies the secant equation. It pays to reexamine the derivation above in the light of this observation.

§6.2 The SR1 Method

It is easy to see that even if B_k is positive definite, B_{k+1} may not have the same property. (The same is, of course, true of H_k.) This observation was considered a major drawback in the early days of nonlinear optimization when only line search iterations were used. However, with the advent of **trust-region** methods, the SR1 updating formula has proved to be quite useful, and its ability to generate indefinite Hessian approximations can actually be regarded as one of its chief advantages.

The main drawback of SR1 updating is that the denominator in (27) or (28) can vanish. In fact, even when the objective function is a convex quadratic, there may be steps on which there is no symmetric rank-1 update that satisfies the secant equation. It pays to reexamine the derivation above in the light of this observation.

§6.2 The SR1 Method

By reasoning in terms of B_k (similar arguments can be applied to H_k), we see that there are three cases:

1. If $(y_k - B_k s_k)^{\mathrm{T}} s_k \neq 0$, then the arguments above show that there is a unique rank-one updating formula satisfying the secant equation, and that it is given by (27).
2. If $y_k = B_k s_k$, then the only updating formula satisfying the secant equation is simply $B_{k+1} = B_k$.
3. If $y_k \neq B_k s_k$ and $(y_k - B_k s_k)^{\mathrm{T}} s_k = 0$, then (26) shows that there is no symmetric rank-one updating formula satisfying the secant equation.

§6.2 The SR1 Method

The last case ($y_k \neq B_k s_k$ and $(y_k - B_k s_k)^{\mathrm{T}} s_k = 0$) clouds an otherwise simple and elegant derivation, and suggests that numerical instabilities and even breakdown of the method can occur. It suggests that rank-one updating does not provide enough freedom to develop a matrix with all the desired characteristics, and that a rank-two correction is required. This reasoning leads us back to the BFGS method, in which positive definiteness (and thus non-singularity) of all Hessian approximations is guaranteed.

§6.2 The SR1 Method

Nevertheless, we are interested in the SR1 formula for the following reasons:

1. A simple safeguard seems to adequately prevent the breakdown of the method and the occurrence of numerical instabilities.
2. The matrices generated by the SR1 formula tend to be good approximations to the true Hessian matrix – often better than the BFGS approximations.
3. In quasi-Newton methods for constrained problems, or in methods for partially separable functions (see Chapters 18 and 7), it may not be possible to impose the curvature condition $y_k^T s_k > 0$, and thus BFGS updating is not recommended. Indeed, in these two settings, indefinite Hessian approximations are desirable insofar as they reflect indefiniteness in the true Hessian.

§6.2 The SR1 Method

Nevertheless, we are interested in the SR1 formula for the following reasons:

1. A simple safeguard seems to adequately prevent the breakdown of the method and the occurrence of numerical instabilities.
2. The matrices generated by the SR1 formula tend to be good approximations to the true Hessian matrix – often better than the BFGS approximations.
3. In quasi-Newton methods for constrained problems, or in methods for partially separable functions (see Chapters 18 and 7), it may not be possible to impose the curvature condition $y_k^{\mathrm{T}} s_k > 0$, and thus BFGS updating is not recommended. Indeed, in these two settings, indefinite Hessian approximations are desirable insofar as they reflect indefiniteness in the true Hessian.

§6.2 The SR1 Method

Nevertheless, we are interested in the SR1 formula for the following reasons:

1. A simple safeguard seems to adequately prevent the breakdown of the method and the occurrence of numerical instabilities.
2. The matrices generated by the SR1 formula tend to be good approximations to the true Hessian matrix – often better than the BFGS approximations.
3. In quasi-Newton methods for constrained problems, or in methods for partially separable functions (see Chapters 18 and 7), it may not be possible to impose the curvature condition $y_k^{\mathrm{T}} s_k > 0$, and thus BFGS updating is not recommended. Indeed, in these two settings, indefinite Hessian approximations are desirable insofar as they reflect indefiniteness in the true Hessian.

§6.2 The SR1 Method

We now introduce a strategy to prevent the SR1 method from breaking down. It has been observed in practice that SR1 performs well simply by skipping the update if the denominator is small. More specifically, the update (27) is applied only if

$$|s_k^{\mathrm{T}}(y_k - B_k s_k)| \geqslant r\,\|s_k\|\|y_k - B_k s_k\|\,, \tag{29}$$

where $r \in (0,1)$ is a small number, say $r = 10^{-8}$. If (29) does not hold, we set $B_{k+1} = B_k$. Most implementations of the SR1 method use a skipping rule of this kind.

§6.2 The SR1 Method

為什麼我們在前一節中不鼓勵在 BFGS 方法的情況下跳過更新，而在 SR1 方法中卻主張跳過更新呢？The two cases are quite different. The condition $s_k^{\mathrm{T}}(y_k - B_k s_k) \approx 0$ occurs infrequently, since it requires certain vectors to be aligned in a specific way. When it does occur, skipping the update appears to have no negative effects on the iteration. This is not surprising, since the skipping condition implies that $s_k^{\mathrm{T}} \bar{G} s_k \approx s_k^{\mathrm{T}} B_k s_k$, where $\bar{G}$ is the average Hessian over the last step – meaning that the curvature of B_k along s_k is already correct. In contrast, the curvature condition $s_k^{\mathrm{T}} y_k \geqslant 0$ required for BFGS updating may easily fail if the line search does not impose the Wolfe conditions (for example, if the step is not long enough), and therefore skipping the BFGS update can occur often and can degrade the quality of the Hessian approximation.

§6.2 The SR1 Method

We now give a formal description of an SR1 method using a trust-region framework, which we prefer over a line search framework because it can accommodate indefinite Hessian approximations more easily.

Algorithm 6.2 (SR1 Trust-Region Method).

Given starting point x_0, initial Hessian approximation B_0, trust-region radius Δ_0, convergence tolerance $\varepsilon > 0$, parameters $\eta \in (0, 10^{-3})$ and $r \in (0, 1)$;

$k \leftarrow 0$;

while $\|\nabla f_k\| > \varepsilon$

Compute s_k by solving the sub-problem

$$\min_s \left[\nabla f_k^{\mathrm{T}} s + \frac{1}{2} s^{\mathrm{T}} B_k s\right] \quad \text{subject to } \|s\| \leqslant \Delta_k; \tag{30}$$

§6.2 The SR1 Method

while $\|\nabla f_k\| > \varepsilon$

Compute s_k by solving the sub-problem

$$\min_s \Big[\nabla f_k^{\mathrm{T}} s + \frac{1}{2} s^{\mathrm{T}} B_k s\Big] \quad \text{subject to } \|s\| \leqslant \Delta_k; \qquad (30)$$

Compute

$$y_k = (\nabla f)(x_k + s_k) - \nabla f_k;$$

$$\text{ared} = f_k - f(x_k + s_k); \qquad \text{(actual reduction)}$$

$$\text{pred} = -\Big(\nabla f_k^{\mathrm{T}} s_k + \frac{1}{2} s_k^{\mathrm{T}} B_k s_k\Big); \qquad \text{(predicted reduction)}$$

if ared/pred $> \eta$

$$x_{k+1} = x_k + s_k;$$

else

$$x_{k+1} = x_k;$$

end (if)

§6.2 The SR1 Method

if ared/pred $> \eta$

$x_{k+1} = x_k + s_k;$

else

$x_{k+1} = x_k;$

end (if)

if ared/pred > 0.75

if $\|s_k\| \leqslant 0.8\Delta_k$

$\Delta_{k+1} = \Delta_k;$

else

$\Delta_{k+1} = 2\Delta_k;$

end (if)

elseif $0.1 \leqslant$ ared/pred $\leqslant 0.75$

$\Delta_{k+1} = \Delta_k;$

else

$\Delta_{k+1} = 0.5\Delta_k;$

end (if)

§6.2 The SR1 Method

elseif $0.1 \leqslant \text{ared}/\text{pred} \leqslant \eta$

$\Delta_{k+1} = \Delta_k$;

else

$\Delta_{k+1} = 0.5\Delta_k$;

end (if)

if $|s_k^{\mathrm{T}}(y_k - B_k s_k)| \geqslant r\|s_k\|\|y_k - B_k s_k\|$

$$B_{k+1} = B_k + \frac{(y_k - B_k s_k)(y_k - B_k s_k)^{\mathrm{T}}}{(y_k - B_k s_k)^{\mathrm{T}} s_k} \text{ (even if } x_{k+1} = x_k\text{);}$$

else

$B_{k+1} \leftarrow B_k$;

end (if)

$k + 1 \leftarrow k$;

end (while)

§6.2 The SR1 Method

This algorithm has the typical form of a trust region method (cf. Algorithm 4.1). For concreteness, we have specified a particular strategy for updating the trust region radius, but other heuristics can be used instead.

To obtain a fast rate of convergence, it is important for the matrix B_k to be updated even along a failed direction s_k. The fact that the step was poor indicates that B_k is an inadequate approximation of the true Hessian in this direction. Unless the quality of the approximation is improved, steps along similar directions could be generated on later iterations, and repeated rejection of such steps could prevent superlinear convergence.

§6.2 The SR1 Method

This algorithm has the typical form of a trust region method (cf. Algorithm 4.1). For concreteness, we have specified a particular strategy for updating the trust region radius, but other heuristics can be used instead.

To obtain a fast rate of convergence, it is important for the matrix B_k to be updated even along a failed direction s_k. The fact that the step was poor indicates that B_k is an inadequate approximation of the true Hessian in this direction. Unless the quality of the approximation is improved, steps along similar directions could be generated on later iterations, and repeated rejection of such steps could prevent superlinear convergence.

§6.2 The SR1 Method

- **Properties of SR1 Updating**

One of the main advantages of SR1 updating is its ability to generate good Hessian approximations. We demonstrate this property by first examining a quadratic function. For functions of this type, the choice of step length does not affect the update, so to examine the effect of the updates, we can assume for simplicity a uniform step length of 1; that is,

$$p_k = -H_k \nabla f_k\,, \quad x_{k+1} = x_k + p_k\,. \tag{31}$$

It follows that $p_k = s_k$.

§6.2 The SR1 Method

Theorem

Suppose that $f:\mathbb{R}^n \to \mathbb{R}$ is the strongly convex quadratic function $f(x) = \frac{1}{2}x^{\mathrm{T}}Qx + b^{\mathrm{T}}x$, where Q is symmetric positive definite. Then for any starting point x_0 and any symmetric starting matrix H_0, the iterates $\{x_k\}$ generated by the SR1 method

$$p_k = -H_k \nabla f_k\,, \quad x_{k+1} = x_k + p_k\,, \tag{31}$$

where H_k satisfies the updating formula

$$\text{(SR1)} \qquad H_{k+1} = H_k + \frac{(s_k - H_k y_k)(s_k - H_k y_k)^{\mathrm{T}}}{(s_k - H_k y_k)^{\mathrm{T}} y_k}\,, \tag{28}$$

converge to the minimizer in at most n steps, provided that $(s_k - H_k y_k)^{\mathrm{T}} y_k \neq 0$ for all k. Moreover, if n steps are performed, and if the search directions p_i are linearly independent, then $H_n = Q^{-1}$.

§6.2 The SR1 Method

Proof.

Because of our assumption $(s_k - H_k y_k)^{\mathrm{T}} y_k \neq 0$, the SR1 update is always well-defined. We start by showing inductively that

$$H_k y_j = s_j \quad \text{for all } j = 0, 1, \cdots, k-1. \tag{32}$$

In other words, we claim that the secant equation is satisfied not only along the most recent search direction, but along all previous directions. By definition, the SR1 update satisfies the secant equation, so we have $H_1 y_0 = s_0$. Therefore, (32) holds for $k = 1$. Let us now assume that (32) holds for some value $k > 1$ and show that it holds also for $k+1$. From this assumption, we have from (32) that

$$(s_k - H_k y_k)^{\mathrm{T}} y_j = s_k^{\mathrm{T}} y_j - y_k^{\mathrm{T}} (H_k y_j) = s_k^{\mathrm{T}} y_j - y_k^{\mathrm{T}} s_j = 0 \quad \forall\, j < k, \tag{33}$$

where the last equality follows because $y_\ell = Q s_\ell$ for the quadratic function we are considering here. □

§6.2 The SR1 Method

Proof.

Because of our assumption $(s_k - H_k y_k)^{\mathrm{T}} y_k \neq 0$, the SR1 update is always well-defined. We start by showing inductively that

$$H_k y_j = s_j \quad \text{for all } j = 0, 1, \cdots, k-1. \tag{32}$$

In other words, we claim that the secant equation is satisfied not only along the most recent search direction, but along all previous directions. By definition, the SR1 update satisfies the secant equation, so we have $H_1 y_0 = s_0$. Therefore, (32) holds for $k = 1$. Let us now assume that (32) holds for some value $k > 1$ and show that it holds also for $k+1$. From this assumption, we have from (32) that

$$(s_k - H_k y_k)^{\mathrm{T}} y_j = s_k^{\mathrm{T}} y_j - y_k^{\mathrm{T}} (H_k y_j) = s_k^{\mathrm{T}} y_j - y_k^{\mathrm{T}} s_j = 0 \quad \forall\, j < k, \tag{33}$$

where the last equality follows because $y_\ell = Q s_\ell$ for the quadratic function we are considering here. □

§6.2 The SR1 Method

Proof.

Because of our assumption $(s_k - H_k y_k)^{\mathrm{T}} y_k \neq 0$, the SR1 update is always well-defined. We start by showing inductively that

$$H_k y_j = s_j \quad \text{for all } j = 0, 1, \cdots, k-1. \tag{32}$$

In other words, we claim that the secant equation is satisfied not only along the most recent search direction, but along all previous directions. By definition, the SR1 update satisfies the secant equation, so we have $H_1 y_0 = s_0$. Therefore, (32) holds for $k = 1$. Let us now assume that (32) holds for some value $k > 1$ and show that it holds also for $k+1$. From this assumption, we have from (32) that

$$(s_k - H_k y_k)^{\mathrm{T}} y_j = s_k^{\mathrm{T}} y_j - y_k^{\mathrm{T}} (H_k y_j) = s_k^{\mathrm{T}} y_j - y_k^{\mathrm{T}} s_j = 0 \quad \forall\, j < k, \tag{33}$$

where the last equality follows because $y_\ell = Q s_\ell$ for the quadratic function we are considering here. □

§6.2 The SR1 Method

Proof (cont'd).

Using (33) and the induction hypothesis (32) in

$$\text{(SR1)} \qquad H_{k+1} = H_k + \frac{(s_k - H_k y_k)(s_k - H_k y_k)^{\mathrm{T}}}{(s_k - H_k y_k)^{\mathrm{T}} y_k}, \tag{28}$$

we have

$$H_{k+1} y_j = H_k y_j = s_j \quad \text{for all } j < k.$$

Since $H_{k+1} y_k = s_k$ by the secant equation, we have shown that (32) holds when k is replaced by $k+1$. By induction, then, this relation holds for all k. If the algorithm performs n steps, and if these steps $\{s_j\}$ are linearly independent, we have

$$s_j = H_n y_j = H_n Q s_j \quad \text{for all } j = 0, 1, \cdots, n-1.$$

It follows that $H_n Q = \mathrm{I}$; that is, $H_n = Q^{-1}$. Therefore, the step taken at x_n is the Newton step, and so the next iterate x_{n+1} will be the solution, and the algorithm terminates. □

§6.2 The SR1 Method

Proof (cont'd).

Using (33) and the induction hypothesis (32) in

$$\text{(SR1)} \qquad H_{k+1} = H_k + \frac{(s_k - H_k y_k)(s_k - H_k y_k)^{\mathrm{T}}}{(s_k - H_k y_k)^{\mathrm{T}} y_k}\,, \tag{28}$$

we have

$$H_{k+1} y_j = H_k y_j = s_j \quad \text{for all } j < k.$$

Since $H_{k+1} y_k = s_k$ by the secant equation, we have shown that (32) holds when k is replaced by $k+1$. By induction, then, this relation holds for all k. If the algorithm performs n steps, and if these steps $\{s_j\}$ are linearly independent, we have

$$s_j = H_n y_j = H_n Q s_j \quad \text{for all } j = 0, 1, \cdots, n-1.$$

It follows that $H_n Q = \mathrm{I}$; that is, $H_n = Q^{-1}$. Therefore, the step taken at x_n is the Newton step, and so the next iterate x_{n+1} will be the solution, and the algorithm terminates. □

§6.2 The SR1 Method

Proof (cont'd).

Using (33) and the induction hypothesis (32) in

$$\text{(SR1)} \qquad H_{k+1} = H_k + \frac{(s_k - H_k y_k)(s_k - H_k y_k)^{\mathrm{T}}}{(s_k - H_k y_k)^{\mathrm{T}} y_k}, \tag{28}$$

we have

$$H_{k+1} y_j = H_k y_j = s_j \quad \text{for all } j < k.$$

Since $H_{k+1} y_k = s_k$ by the secant equation, we have shown that (32) holds when k is replaced by $k+1$. By induction, then, this relation holds for all k. If the algorithm performs n steps, and if these steps $\{s_j\}$ are linearly independent, we have

$$s_j = H_n y_j = H_n Q s_j \quad \text{for all } j = 0, 1, \cdots, n-1.$$

It follows that $H_n Q = \mathrm{I}$; that is, $H_n = Q^{-1}$. Therefore, the step taken at x_n is the Newton step, and so the next iterate x_{n+1} will be the solution, and the algorithm terminates. □

§6.2 The SR1 Method

Proof (cont'd).

Consider now the case in which the steps become linearly dependent. Suppose that s_k is a linear combination of the previous steps:

$$s_k = \xi_0 s_0 + \cdots + \xi_{k-1} s_{k-1} ,$$

for some scalars $\xi_0, \cdots, \xi_{k-1}$. From (32) we have that

$$\begin{aligned} H_k y_k &= H_k Q s_k = \xi_0 H_k Q s_0 + \cdots + \xi_{k-1} H_k Q s_{k-1} \\ &= \xi_0 H_k y_0 + \cdots + \xi_{k-1} H_k y_{k-1} \\ &= \xi_0 s_0 + \cdots + \xi_{k-1} s_{k-1} = s_k . \end{aligned}$$

Since $y_k = \nabla f_{k+1} - \nabla f_k$ and since $s_k = p_k = -H_k \nabla f_k$ from (31), we have that

$$H_k(\nabla f_{k+1} - \nabla f_k) = -H_k \nabla f_k ,$$

which, by the non-singularity of H_k, implies that $\nabla f_{k+1} = 0$. Therefore, x_{k+1} is the solution point. □

§6.2 The SR1 Method

Proof (cont'd).

Consider now the case in which the steps become linearly dependent. Suppose that s_k is a linear combination of the previous steps:

$$s_k = \xi_0 s_0 + \cdots + \xi_{k-1} s_{k-1} ,$$

for some scalars $\xi_0, \cdots, \xi_{k-1}$. From (32) we have that

$$\begin{aligned} H_k y_k &= H_k Q s_k = \xi_0 H_k Q s_0 + \cdots + \xi_{k-1} H_k Q s_{k-1} \\ &= \xi_0 H_k y_0 + \cdots + \xi_{k-1} H_k y_{k-1} \\ &= \xi_0 s_0 + \cdots + \xi_{k-1} s_{k-1} = s_k . \end{aligned}$$

Since $y_k = \nabla f_{k+1} - \nabla f_k$ and since $s_k = p_k = -H_k \nabla f_k$ from (31), we have that

$$H_k(\nabla f_{k+1} - \nabla f_k) = -H_k \nabla f_k ,$$

which, by the non-singularity of H_k, implies that $\nabla f_{k+1} = 0$. Therefore, x_{k+1} is the solution point. □

§6.2 The SR1 Method

The relation (32) shows that when f is quadratic, the secant equation is satisfied along all previous search directions, regardless of how the line search is performed. A result like this can be established for BFGS updating only under the restrictive assumption that the line search is exact, as we show in the next section.

§6.2 The SR1 Method

For general nonlinear functions, the SR1 update continues to generate good Hessian approximations under certain conditions. Before stating the last theorem in this section, we need to talk about the **uniform linear independence** of a sequence.

Definition

A sequence of vectors $\{x_k\} \subseteq \mathbb{R}^n$ is said to be uniformly linearly independent if there exist integers $m \geqslant n$, $k_0 \geqslant 0$ and a constant $c > 0$ such that, for each $k \geqslant k_0$,

$$\max\left\{\left|\frac{\langle x, x_{k+j}\rangle}{\|x\|\|x_{k+j}\|}\right| \,\middle|\, j = 1, \cdots, m\right\} \geqslant c \quad \forall\, x \in \mathbb{R}^n\,.$$

In other words, the uniform linear independence of a sequence means that, up to deleting the first few terms from the sequence, any consecutive m terms, where $m \geqslant n$, span $\mathbb{R}^n$ in a "certain" manner.

§6.2 The SR1 Method

Theorem

Suppose that $f : \mathbb{R}^n \to \mathbb{R}$ is twice continuously differentiable, and that its Hessian is bounded and Lipschitz continuous in a neighborhood of a point x_*. Let $\{x_k\}$ be any sequence of iterates converging to x_*. Suppose in addition that for some $r \in (0,1)$ the inequality

$$|s_k^{\mathrm{T}}(y_k - B_k s_k)| \geqslant r \|s_k\| \|y_k - B_k s_k\| \,, \tag{29}$$

holds for all k, and that the steps s_k are uniformly linearly independent. Then the matrices B_k generated by the SR1 updating formula satisfy

$$\lim_{k\to\infty} \|B_k - (\nabla^2 f)(x_*)\| = 0 \,.$$

§6.3 The Broyden Class

So far, we have described the BFGS, DFP, and SR1 quasi-Newton updating formulae, but there are many others. Of particular interest is the **Broyden class**, a family of updates specified by the following general formula:

$$B_{k+1} = B_k - \frac{B_k s_k s_k^{\mathrm{T}} B_k}{s_k^{\mathrm{T}} B_k s_k} + \frac{y_k y_k^{\mathrm{T}}}{y_k^{\mathrm{T}} s_k} + \phi_k (s_k^{\mathrm{T}} B_k s_k) v_k v_k^{\mathrm{T}} , \tag{34}$$

where ϕ_k is a scalar parameter and

$$v_k = \left[\frac{y_k}{y_k^{\mathrm{T}} s_k} - \frac{B_k s_k}{s_k^{\mathrm{T}} B_k s_k} \right] . \tag{35}$$

§6.3 The Broyden Class

The BFGS and DFP methods are members of the Broyden class – we recover BFGS by setting $\phi_k = 0$ and DFP by setting $\phi_k = 1$ in (34). We can therefore rewrite (34) as a "linear combination" (the exact terminology is affine combination) of these two methods; that is,

$$B_{k+1} = (1-\phi_k)B_{k+1}^{\text{BFGS}} + \phi_k B_{k+1}^{\text{DFP}}\,.$$

This relationship indicates that all members of the Broyden class satisfy the secant equation (6), since the BFGS and DFP matrices themselves satisfy this equation. Also, since BFGS and DFP updating preserve positive definiteness of the Hessian approximations when $s_k^{\mathrm{T}} y_k > 0$, this relation implies that the same property will hold for the Broyden family if $0 \leqslant \phi_k \leqslant 1$.

§6.3 The Broyden Class

The BFGS and DFP methods are members of the Broyden class – we recover BFGS by setting $\phi_k = 0$ and DFP by setting $\phi_k = 1$ in (34). We can therefore rewrite (34) as a "linear combination" (the exact terminology is affine combination) of these two methods; that is,

$$B_{k+1} = (1 - \phi_k) B_{k+1}^{\text{BFGS}} + \phi_k B_{k+1}^{\text{DFP}} .$$

This relationship indicates that all members of the Broyden class satisfy the secant equation (6), since the BFGS and DFP matrices themselves satisfy this equation. Also, since BFGS and DFP updating preserve positive definiteness of the Hessian approximations when $s_k^{\mathrm{T}} y_k > 0$, this relation implies that the same property will hold for the Broyden family if $0 \leqslant \phi_k \leqslant 1$.

§6.3 The Broyden Class

Much attention has been given to the so-called ***restricted*** Broyden class, which is obtained by restricting ϕ_k to the interval $[0, 1]$. It enjoys the following property when applied to quadratic functions. Since the analysis is independent of the step length, we assume for simplicity that each iteration has the form

$$p_k = -B_k^{-1}\nabla f_k\,, \quad x_{k+1} = x_k + p_k\,. \tag{36}$$

§6.3 The Broyden Class

Theorem

Suppose that $f:\mathbb{R}^n \to \mathbb{R}$ is the strongly convex quadratic function $f(x) = \frac{1}{2}x^{\mathrm{T}}Qx + b^{\mathrm{T}}x$, where Q is symmetric and positive definite. Let x_0 be any starting point for the iteration (36) and B_0 be any symmetric positive definite starting matrix, and suppose that the matrices B_k are updated by the Broyden formula (34) with $\phi_k \in [0,1]$. Define $\lambda_1^{(k)} \leqslant \cdots \leqslant \lambda_n^{(k)}$ to be the eigenvalues of the matrix

$$Q^{1/2}B_k^{-1}Q^{1/2}\,. \tag{37}$$

Then for all k, we have

$$\min\big\{\lambda_j^{(k)},1\big\} \leqslant \lambda_j^{(k+1)} \leqslant \max\big\{\lambda_j^{(k)},1\big\} \quad \text{for } j=1,2,\cdots,n. \tag{38}$$

Moreover, the property (38) does not hold if the Broyden parameter ϕ_k is chosen outside the interval $[0,1]$.

§6.3 The Broyden Class

讓我們約略說明一下這個結果的重要性。如果矩陣 $Q^{1/2}B_k^{-1}Q^{1/2}$ 的特徵值 $\lambda_i^{(k)}$ 都是 1，那麼 quasi-Newton 方法中用來逼近 Hessian 的矩陣 B_k 將與二次目標函數的 Hessian 矩陣 Q 相同。雖說這是理想情況，但我們會因此希望 $Q^{1/2}B_k^{-1}Q^{1/2}$ 的特徵值越接近 1 越好。事實上，(38) 式告訴我們 $Q^{1/2}B_k^{-1}Q^{1/2}$ 的特徵值 $\{\lambda_i^{(k)}\}$ 在 k 趨近 ∞ 時是收斂到 1。例如，假設在第 k 次迭代時最小的特徵值為 0.7。那麼，根據 (38) 式，在下一次迭代中，特徵值將落在 $[0.7, 1]$ 的範圍內。雖然我們無法確定這個特徵值是否實際上已經更接近 1，但可以合理地期望它已經更接近 1。相比之下，如果我們允許 ϕ_k 超出 $[0, 1]$，第一個特徵值可能會變得小於 0.7。值得注意的是，即使在進行 line search 時不是 exact，該定理的結果仍然成立。

§6.3 The Broyden Class

讓我們約略說明一下這個結果的重要性。如果矩陣 $Q^{1/2}B_k^{-1}Q^{1/2}$ 的特徵值 $\lambda_i^{(k)}$ 都是 1，那麼 quasi-Newton 方法中用來逼近 Hessian 的矩陣 B_k 將與二次目標函數的 Hessian 矩陣 Q 相同。雖說這是理想情況，但我們會因此希望 $Q^{1/2}B_k^{-1}Q^{1/2}$ 的特徵值越接近 1 越好。事實上，(38) 式告訴我們 $Q^{1/2}B_k^{-1}Q^{1/2}$ 的特徵值 $\{\lambda_i^{(k)}\}$ 在 k 趨近 ∞ 時是收斂到 1。例如，假設在第 k 次迭代時最小的特徵值為 0.7。那麼，根據 (38) 式，在下一次迭代中，特徵值將落在 $[0.7, 1]$ 的範圍內。雖然我們無法確定這個特徵值是否實際上已經更接近 1，但可以合理地期望它已經更接近 1。相比之下，如果我們允許 ϕ_k 超出 $[0, 1]$，第一個特徵值可能會變得小於 0.7。值得注意的是，即使在進行 line search 時不是 exact，該定理的結果仍然成立。

§6.3 The Broyden Class

Although the theorem seems to suggest that the best update formulas belong to the restricted Broyden class, the situation is not at all clear. Some analysis and computational testing suggest that algorithms that allow ϕ_k to be negative (in a strictly controlled manner) may in fact be superior to the BFGS method. The SR1 formula is a case in point: It is a member of the Broyden class, obtained by setting

$$\phi_k = \frac{s_k^{\mathrm{T}} y_k}{s_k^{\mathrm{T}} y_k - s_k^{\mathrm{T}} B_k s_k},$$

but it does not belong to the restricted Broyden class, because this value of ϕ_k may fall outside the interval $[0, 1]$.

§6.3 The Broyden Class

Although the theorem seems to suggest that the best update formulas belong to the restricted Broyden class, the situation is not at all clear. Some analysis and computational testing suggest that algorithms that allow ϕ_k to be negative (in a strictly controlled manner) may in fact be superior to the BFGS method. The SR1 formula is a case in point: It is a member of the Broyden class, obtained by setting

$$\phi_k = \frac{s_k^{\mathrm{T}} y_k}{s_k^{\mathrm{T}} y_k - s_k^{\mathrm{T}} B_k s_k},$$

but it does not belong to the restricted Broyden class, because this value of ϕ_k may fall outside the interval $[0, 1]$.

§6.3 The Broyden Class

In the remaining discussion of this section, we determine more precisely the range of values of ϕ_k that preserve positive definiteness. The last term in

$$B_{k+1} = B_k - \frac{B_k s_k s_k^{\mathrm{T}} B_k}{s_k^{\mathrm{T}} B_k s_k} + \frac{y_k y_k^{\mathrm{T}}}{y_k^{\mathrm{T}} s_k} + \phi_k (s_k^{\mathrm{T}} B_k s_k) v_k v_k^{\mathrm{T}} \tag{34}$$

is a rank-one correction, which by the **interlacing eigenvalue theorem** (in the next slide) increases the eigenvalues of the matrix when ϕ_k is positive. Therefore, B_{k+1} is positive definite for all $\phi_k \geqslant 0$. On the other hand, by the interlacing eigenvalue theorem the last term in (34) decreases the eigenvalues of the matrix when ϕ_k is negative. As we decrease ϕ_k, this matrix eventually becomes singular and then indefinite.

§6.3 The Broyden Class

Theorem (Interlacing Eigenvalue Theorem)

Let $A \in \mathbb{R}^{n\times n}$ be a symmetric matrix with eigenvalues $\lambda_1, \lambda_2, \cdots, \lambda_n$ satisfying $\lambda_1 \leqslant \lambda_2 \leqslant \cdots \leqslant \lambda_n$, and let $z \in \mathbb{R}^n$ be a vector with $\|z\| = 1$, and $\alpha \in \mathbb{R}$ be a scalar. Then if we denote the eigenvalues of $A + \alpha zz^{\mathrm{T}}$ by $\xi_1, \xi_2, \cdots, \xi_n$ (in increasing order), we have for $\alpha > 0$ that

$$\lambda_1 \leqslant \xi_1 \leqslant \lambda_2 \leqslant \xi_2 \leqslant \cdots \leqslant \lambda_n \leqslant \xi_n ,$$

with

$$\sum_{i=1}^{n} (\xi_i - \lambda_i) = \alpha . \tag{39}$$

If $\alpha < 0$, we have that

$$\xi_1 \leqslant \lambda_1 \leqslant \xi_2 \leqslant \lambda_2 \leqslant \cdots \leqslant \xi_n \leqslant \lambda_n ,$$

where the relationship (39) is again satisfied.

§6.3 The Broyden Class

In the remaining discussion of this section, we determine more precisely the range of values of ϕ_k that preserve positive definiteness. The last term in

$$B_{k+1} = B_k - \frac{B_k s_k s_k^{\mathrm{T}} B_k}{s_k^{\mathrm{T}} B_k s_k} + \frac{y_k y_k^{\mathrm{T}}}{y_k^{\mathrm{T}} s_k} + \phi_k (s_k^{\mathrm{T}} B_k s_k) v_k v_k^{\mathrm{T}} \tag{34}$$

is a rank-one correction, which by the **interlacing eigenvalue theorem** (in the next slide) increases the eigenvalues of the matrix when ϕ_k is positive. Therefore, B_{k+1} is positive definite for all $\phi_k \geqslant 0$. On the other hand, by the interlacing eigenvalue theorem the last term in (34) decreases the eigenvalues of the matrix when ϕ_k is negative. As we decrease ϕ_k, this matrix eventually becomes singular and then indefinite.

§6.3 The Broyden Class

A little computation shows that B_{k+1} is singular when ϕ_k has the value

$$\phi_k^c = \frac{1}{1-\mu_k}, \tag{40}$$

where

$$\mu_k = \frac{(y_k^{\mathrm{T}} B_k^{-1} y_k)(s_k^{\mathrm{T}} B_k s_k)}{(y_k^{\mathrm{T}} s_k)^2}. \tag{41}$$

By applying the Cauchy-Schwarz inequality to (41), we see that $\mu_k \geqslant 1$ and therefore $\phi_k^c \leqslant 0$. Hence, if the initial Hessian approximation B_0 is symmetric and positive definite, and if $s_k^{\mathrm{T}} y_k > 0$ and $\phi_k > \phi_k^c$ for each k, then all the matrices B_k generated by Broyden's formula (34) remain symmetric and positive definite.

§6.3 The Broyden Class

When the line search is exact, all methods in the Broyden class with $\phi_k \geqslant \phi_k^c$ generate the same sequence of iterates. This result applies to general nonlinear functions and is based on the observation that when all the line searches are exact, the directions generated by Broyden-class methods differ only in their lengths. The line searches identify the same minima along the chosen search direction, though the values of the step lengths may differ because of the different scaling.

The Broyden class has several remarkable properties when applied with exact line searches to quadratic functions. We state some of these properties in the next theorem, whose proof is omitted.

§6.3 The Broyden Class

Theorem

Suppose that a method in the Broyden class is applied to the strongly convex quadratic function $f(x) = b^{\mathrm{T}}x + \frac{1}{2}x^{\mathrm{T}}Qx$, *where* x_0 *is the starting point and* B_0 *is any symmetric positive definite matrix. Assume that* α_k *is the exact step length and that* $\phi_k \geqslant \phi_k^c$ *for all* k, *where* ϕ_k^c *is defined by*

$$\phi_k^c = \frac{1}{1-\mu_k}, \quad \mu_k = \frac{(y_k^{\mathrm{T}} B_k^{-1} y_k)(s_k^{\mathrm{T}} B_k s_k)}{(y_k^{\mathrm{T}} s_k)^2}.$$

Then the following statements are true.

1. *The iterates are independent of* ϕ_k *and converge to the solution in at most* n *iterations.*
2. *The secant equation is satisfied for all previous search directions; that is,* $B_k s_j = y_j$ *for* $j = 1, 2, \cdots, k-1$.

§6.3 The Broyden Class

Theorem (cont'd)

3. *If the starting matrix is $B_0 = \mathrm{I}$, then the iterates are identical to those generated by the conjugate gradient method. In particular, the search directions are conjugate; that is,*

$$s_i^{\mathrm{T}} Q s_j = 0 \quad \text{for } i \neq j.$$

4. *If n iterations are performed, we have $B_n = Q$.*

Note that parts ①, ②, and ④ of this result echo the statement and proof of the theorem in Section 6.2, where similar results were derived for the SR1 update formula.

§6.3 The Broyden Class

We can generalize the theorem slightly: It continues to hold if the Hessian approximations remain non-singular but not necessarily positive definite. (Hence, we could allow ϕ_k to be smaller than ϕ_k^c, provided that the chosen value did not produce a singular updated matrix.) We can also generalize point ③ as follows. If the starting matrix B_0 is not the identity matrix, then the Broyden-class method is identical to the preconditioned conjugate gradient method that uses B_0 as preconditioner.

We conclude by commenting that results like the theorem would appear to be of mainly theoretical interest, since the inexact line searches used in practical implementations of Broyden-class methods (and all other quasi-Newton methods) cause their performance to differ markedly. Nevertheless, it is worth noting that this type of analysis guided much of the development of quasi-Newton methods.

§6.3 The Broyden Class

We can generalize the theorem slightly: It continues to hold if the Hessian approximations remain non-singular but not necessarily positive definite. (Hence, we could allow ϕ_k to be smaller than ϕ_k^c, provided that the chosen value did not produce a singular updated matrix.) We can also generalize point ③ as follows. If the starting matrix B_0 is not the identity matrix, then the Broyden-class method is identical to the preconditioned conjugate gradient method that uses B_0 as preconditioner.

We conclude by commenting that results like the theorem would appear to be of mainly theoretical interest, since the inexact line searches used in practical implementations of Broyden-class methods (and all other quasi-Newton methods) cause their performance to differ markedly. Nevertheless, it is worth noting that this type of analysis guided much of the development of quasi-Newton methods.

§6.3 The Broyden Class

We can generalize the theorem slightly: It continues to hold if the Hessian approximations remain non-singular but not necessarily positive definite. (Hence, we could allow ϕ_k to be smaller than ϕ_k^c, provided that the chosen value did not produce a singular updated matrix.) We can also generalize point ③ as follows. If the starting matrix B_0 is not the identity matrix, then the Broyden-class method is identical to the preconditioned conjugate gradient method that uses B_0 as preconditioner.

We conclude by commenting that results like the theorem would appear to be of mainly theoretical interest, since the inexact line searches used in practical implementations of Broyden-class methods (and all other quasi-Newton methods) cause their performance to differ markedly. Nevertheless, it is worth noting that this type of analysis guided much of the development of quasi-Newton methods.

§6.4 Convergence Analysis

In this section we present global and local convergence results for practical implementations of the BFGS and SR1 methods. We give more details for BFGS because its analysis is more general and illuminating than that of SR1. The fact that the Hessian approximations evolve by means of updating formulas makes the analysis of quasi-Newton methods much more complex than that of steepest descent and Newton's method.

§6.4 Convergence Analysis

Although the BFGS and SR1 methods are known to be remarkably robust in practice, we will not be able to establish truly **global** convergence results for general nonlinear objective functions; that is, we **cannot** prove that the iterates of these quasi-Newton methods approach a stationary point of the problem from any starting point and any (suitable) initial Hessian approximation. In fact, it is not yet known if the algorithms enjoy such properties. In our analysis we will either assume that the objective function is convex or that the iterates satisfy certain properties. On the other hand, there are well known local, superlinear convergence results that are true under reasonable assumptions.

Throughout this section we use $\|\cdot\|$ to denote the Euclidean vector or matrix norm, and sometimes denote the Hessian $(\nabla^2 f)(x)$ by $G(x)$.

§6.4 Convergence Analysis

Although the BFGS and SR1 methods are known to be remarkably robust in practice, we will not be able to establish truly **global** convergence results for general nonlinear objective functions; that is, we **cannot** prove that the iterates of these quasi-Newton methods approach a stationary point of the problem from any starting point and any (suitable) initial Hessian approximation. In fact, it is not yet known if the algorithms enjoy such properties. In our analysis we will either assume that the objective function is convex or that the iterates satisfy certain properties. On the other hand, there are well known local, superlinear convergence results that are true under reasonable assumptions.

Throughout this section we use $\|\cdot\|$ to denote the Euclidean vector or matrix norm, and sometimes denote the Hessian $(\nabla^2 f)(x)$ by $G(x)$.

§6.4 Convergence Analysis

- **Global Convergence of the BFGS Method**

We study the global convergence of the BFGS method, with a practical line search, when applied to a smooth convex function from an arbitrary starting point x_0 and from any initial Hessian approximation B_0 that is symmetric and positive definite. We state our precise assumptions about the objective function formally, as follows.

Assumption 6.1.

There exists a convex set C such that

1. The level set $S = \{x \in \mathbb{R}^n \mid f(x) \leqslant f(x_0)\}$ is contained inside C.
2. The objective function f is twice continuously differentiable on C, and there exist positive constants m and M such that

$$m\|z\|^2 \leqslant z^{\mathrm{T}}(\nabla^2 f)(x)z \leqslant M\|z\|^2 \quad \forall\, z \in \mathbb{R}^n, x \in C. \qquad (42)$$

§6.4 Convergence Analysis

- **Global Convergence of the BFGS Method**

We study the global convergence of the BFGS method, with a practical line search, when applied to a smooth convex function from an arbitrary starting point x_0 and from any initial Hessian approximation B_0 that is symmetric and positive definite. We state our precise assumptions about the objective function formally, as follows.

Assumption 6.1.

There exists a convex set C such that

1. The level set $S = \left\{x \in \mathbb{R}^n \,\middle|\, f(x) \leqslant f(x_0)\right\}$ is contained inside C.
2. The objective function f is twice continuously differentiable on C, and there exist positive constants m and M such that

$$m\|z\|^2 \leqslant z^{\mathrm{T}}(\nabla^2 f)(x)z \leqslant M\|z\|^2 \quad \forall\, z \in \mathbb{R}^n, x \in C. \tag{42}$$

§6.4 Convergence Analysis

Part ② of this assumption implies that the Hessian $\nabla^2 f$ is positive definite on S and that f has a unique minimizer x_* in S.

Recall the identity $y_k = \bar{G}_k \alpha_k p_k = \bar{G}_k s_k$, where $\bar{G}_k$ is the average Hessian defined in

$$\bar{G}_k = \left[\int_0^1 (\nabla^2 f)(x_k + \tau\alpha_k p_k)\, d\tau\right]. \tag{11}$$

Using this identity above and (42), we obtain

$$\frac{y_k^{\mathrm{T}} s_k}{s_k^{\mathrm{T}} s_k} = \frac{s_k^{\mathrm{T}} \bar{G}_k s_k}{s_k^{\mathrm{T}} s_k} \geqslant m\,. \tag{43}$$

Assumption 6.1 implies that $\bar{G}_k$ is positive definite, so its square root is well-defined. Therefore, by defining $z_k = \bar{G}_k^{1/2} s_k$,

$$\frac{y_k^{\mathrm{T}} y_k}{y_k^{\mathrm{T}} s_k} = \frac{s_k^{\mathrm{T}} \bar{G}_k^2 s_k}{s_k^{\mathrm{T}} \bar{G}_k s_k} = \frac{z_k^{\mathrm{T}} \bar{G}_k z_k}{z_k^{\mathrm{T}} z_k} \leqslant M\,. \tag{44}$$

§6.4 Convergence Analysis

Part ② of this assumption implies that the Hessian $\nabla^2 f$ is positive definite on S and that f has a unique minimizer x_* in S.

Recall the identity $y_k = \bar{G}_k \alpha_k p_k = \bar{G}_k s_k$, where $\bar{G}_k$ is the average Hessian defined in

$$\bar{G}_k = \left[\int_0^1 (\nabla^2 f)(x_k + \tau \alpha_k p_k)\, d\tau \right]. \tag{11}$$

Using this identity above and (42), we obtain

$$\frac{y_k^{\mathrm{T}} s_k}{s_k^{\mathrm{T}} s_k} = \frac{s_k^{\mathrm{T}} \bar{G}_k s_k}{s_k^{\mathrm{T}} s_k} \geqslant m. \tag{43}$$

Assumption 6.1 implies that $\bar{G}_k$ is positive definite, so its square root is well-defined. Therefore, by defining $z_k = \bar{G}_k^{1/2} s_k$,

$$\frac{y_k^{\mathrm{T}} y_k}{y_k^{\mathrm{T}} s_k} = \frac{s_k^{\mathrm{T}} \bar{G}_k^2 s_k}{s_k^{\mathrm{T}} \bar{G}_k s_k} = \frac{z_k^{\mathrm{T}} \bar{G}_k z_k}{z_k^{\mathrm{T}} z_k} \leqslant M. \tag{44}$$

§6.4 Convergence Analysis

Part ② of this assumption implies that the Hessian $\nabla^2 f$ is positive definite on S and that f has a unique minimizer x_* in S.

Recall the identity $y_k = \bar{G}_k \alpha_k p_k = \bar{G}_k s_k$, where $\bar{G}_k$ is the average Hessian defined in

$$\bar{G}_k = \left[\int_0^1 (\nabla^2 f)(x_k + \tau\alpha_k p_k)\, d\tau\right]. \tag{11}$$

Using this identity above and (42), we obtain

$$\frac{y_k^{\mathrm{T}} s_k}{s_k^{\mathrm{T}} s_k} = \frac{s_k^{\mathrm{T}} \bar{G}_k s_k}{s_k^{\mathrm{T}} s_k} \geqslant m\,. \tag{43}$$

Assumption 6.1 implies that $\bar{G}_k$ is positive definite, so its square root is well-defined. Therefore, by defining $z_k = \bar{G}_k^{1/2} s_k$,

$$\frac{y_k^{\mathrm{T}} y_k}{y_k^{\mathrm{T}} s_k} = \frac{s_k^{\mathrm{T}} \bar{G}_k^2 s_k}{s_k^{\mathrm{T}} \bar{G}_k s_k} = \frac{z_k^{\mathrm{T}} \bar{G}_k z_k}{z_k^{\mathrm{T}} z_k} \leqslant M\,. \tag{44}$$

§6.4 Convergence Analysis

Theorem

Let B_0 be any symmetric positive definite initial matrix, and let x_0 be a starting point for which Assumption 6.1 is satisfied. Then the sequence $\{x_k\}$ generated by Algorithm 6.1 (with $\varepsilon = 0$) converges to the minimizer x_* of f.

Proof.

Let θ_k be the angle between the steepest descent direction and the search direction $p_k = -B_k^{-1}\nabla f_k$. We first prove that $\liminf\limits_{k\to\infty} \|\nabla f_k\| = 0$, using Zoutendijk's condition

$$\sum_{k=0}^{\infty} \cos^2\theta_k \|\nabla f_k\|^2 < \infty \quad \Big(\Rightarrow \lim_{k\to\infty} \cos^2\theta_k \|\nabla f_k\|^2 = 0\Big),$$

by showing that there exist $\delta > 0$ such that

$$\#\big\{k \in \mathbb{N} \,\big|\, |\cos\theta_k| \geqslant \delta\big\} = \infty\,. \qquad \square$$

§6.4 Convergence Analysis

Theorem

Let B_0 be any symmetric positive definite initial matrix, and let x_0 be a starting point for which Assumption 6.1 is satisfied. Then the sequence $\{x_k\}$ generated by Algorithm 6.1 (with $\varepsilon = 0$) converges to the minimizer x_* of f.

Proof.

Let θ_k be the angle between the steepest descent direction and the search direction $p_k = -B_k^{-1}\nabla f_k$. We first prove that $\liminf\limits_{k\to\infty} \|\nabla f_k\| = 0$, using Zoutendijk's condition

$$\sum_{k=0}^{\infty} \cos^2\theta_k \|\nabla f_k\|^2 < \infty \quad \Big(\Rightarrow \lim_{k\to\infty} \cos^2\theta_k \|\nabla f_k\|^2 = 0\Big),$$

by showing that there exist $\delta > 0$ such that

$$\#\Big\{k \in \mathbb{N} \,\Big|\, |\cos\theta_k| \geqslant \delta\Big\} = \infty\,.$$

□

§6.4 Convergence Analysis

Proof (cont'd).

We first compute $\det(B_{k+1})$ in terms of $\det(B_k)$. Since B_k is positive definite, $B_k = P_k \Lambda_k P_k^{\mathrm{T}}$ for some orthogonal matrix P_k and diagonal matrix Λ_k. Using the BFGS updating formula

$$\text{(BFGS)} \qquad B_{k+1} = B_k - \frac{B_k s_k s_k^{\mathrm{T}} B_k}{s_k^{\mathrm{T}} B_k s_k} + \frac{y_k y_k^{\mathrm{T}}}{y_k^{\mathrm{T}} s_k}\,. \tag{22}$$

we find that

$$\Lambda_k^{-1/2} P_k^{\mathrm{T}} B_{k+1} P_k \Lambda_k^{-1/2} = \mathrm{I} - \frac{\eta_k \eta_k^{\mathrm{T}}}{\|\eta_k\|^2} + \frac{w_k w_k^{\mathrm{T}}}{y_k^{\mathrm{T}} s_k}\,,$$

where $\eta_k = \Lambda_k^{1/2} P_k^{\mathrm{T}} s_k$ and $w_k = \Lambda_k^{-1/2} P_k^{\mathrm{T}} y_k$. Let Q_k be an orthogonal matrix satisfying $Q_k \frac{\eta_k}{\|\eta_k\|} = \mathrm{e}_n$, and define $v_k = Q_k w_k$. Then

$$Q_k \Lambda_k^{-1/2} P_k^{\mathrm{T}} B_{k+1} P_k \Lambda_k^{-1/2} Q_k^{\mathrm{T}} = \mathrm{I} - \mathrm{e}_n \mathrm{e}_n^{\mathrm{T}} + \frac{v_k v_k^{\mathrm{T}}}{y_k^{\mathrm{T}} s_k}\,. \qquad \square$$

§6.4 Convergence Analysis

Proof (cont'd).

We first compute $\det(B_{k+1})$ in terms of $\det(B_k)$. Since B_k is positive definite, $B_k = P_k \Lambda_k P_k^{\mathrm{T}}$ for some orthogonal matrix P_k and diagonal matrix Λ_k. Using the BFGS updating formula

$$\text{(BFGS)} \qquad B_{k+1} = B_k - \frac{B_k s_k s_k^{\mathrm{T}} B_k}{s_k^{\mathrm{T}} B_k s_k} + \frac{y_k y_k^{\mathrm{T}}}{y_k^{\mathrm{T}} s_k}\,. \tag{22}$$

we find that

$$\Lambda_k^{-1/2} P_k^{\mathrm{T}} B_{k+1} P_k \Lambda_k^{-1/2} = \mathrm{I} - \frac{\eta_k \eta_k^{\mathrm{T}}}{\|\eta_k\|^2} + \frac{w_k w_k^{\mathrm{T}}}{y_k^{\mathrm{T}} s_k}\,,$$

where $\eta_k = \Lambda_k^{1/2} P_k^{\mathrm{T}} s_k$ and $w_k = \Lambda_k^{-1/2} P_k^{\mathrm{T}} y_k$. Let Q_k be an orthogonal matrix satisfying $Q_k \dfrac{\eta_k}{\|\eta_k\|} = \mathrm{e}_n$, and define $v_k = Q_k w_k$. Then

$$Q_k \Lambda_k^{-1/2} P_k^{\mathrm{T}} B_{k+1} P_k \Lambda_k^{-1/2} Q_k^{\mathrm{T}} = \mathrm{I} - \mathrm{e}_n \mathrm{e}_n^{\mathrm{T}} + \frac{v_k v_k^{\mathrm{T}}}{y_k^{\mathrm{T}} s_k}\,. \qquad \square$$

§6.4 Convergence Analysis

Proof (cont'd).

Suppose that $v_k = [a_1, a_2, \cdots, a_n]^{\mathrm{T}}$. Then

$$y_k^{\mathrm{T}} s_k = w_k^{\mathrm{T}} \eta_k = (Q_k w_k)^{\mathrm{T}} (Q_k \eta_k) = v_k^{\mathrm{T}} \|\eta_k\| \mathrm{e}_n = \|\eta_k\| a_n$$

so that $a_n \neq 0$. Therefore, the matrix $\mathrm{I} - \mathrm{e}_n \mathrm{e}_n^{\mathrm{T}} + \dfrac{v_k v_k^{\mathrm{T}}}{y_k^{\mathrm{T}} s_k}$ is given by

$$\begin{bmatrix} 1 + \dfrac{a_1}{a_n}\dfrac{a_1}{\|\eta_k\|} & \dfrac{a_2}{a_n}\dfrac{a_1}{\|\eta_k\|} & \cdots & \cdots & \dfrac{a_{n-1}}{a_n}\dfrac{a_1}{\|\eta_k\|} & \dfrac{a_1}{\|\eta_k\|} \\ \dfrac{a_1}{a_n}\dfrac{a_2}{\|\eta_k\|} & 1 + \dfrac{a_2}{a_n}\dfrac{a_2}{\|\eta_k\|} & \dfrac{a_3}{a_n}\dfrac{a_2}{\|\eta_k\|} & \cdots & \dfrac{a_{n-1}}{a_n}\dfrac{a_2}{\|\eta_k\|} & \dfrac{a_2}{\|\eta_k\|} \\ \vdots & & \ddots & & & \vdots \\ \vdots & & & \ddots & & \vdots \\ \dfrac{a_1}{a_n}\dfrac{a_{n-1}}{\|\eta_k\|} & & & & 1 + \dfrac{a_{n-1}}{a_n}\dfrac{a_{n-1}}{\|\eta_k\|} & \dfrac{a_{n-1}}{\|\eta_k\|} \\ \dfrac{a_1}{a_n}\dfrac{a_n}{\|\eta_k\|} & \dfrac{a_2}{a_n}\dfrac{a_n}{\|\eta_k\|} & \cdots & \cdots & \dfrac{a_{n-1}}{a_n}\dfrac{a_n}{\|\eta_k\|} & \dfrac{a_n}{\|\eta_k\|} \end{bmatrix}.$$

$\square$

§6.4 Convergence Analysis

Proof (cont'd).

Note that $\eta_k = \Lambda_k^{1/2} P_k^{\mathrm{T}} s_k$ so that

$$\|\eta_k\|^2 = \eta_k^{\mathrm{T}} \eta_k = s_k^{\mathrm{T}} P_k \Lambda_k P_k^{\mathrm{T}} s_k = s_k^{\mathrm{T}} B_k s_k \,.$$

Using the properties of determinants,

$$\det\Big(\mathrm{I} - \mathrm{e}_n \mathrm{e}_n^{\mathrm{T}} + \frac{v_k v_k^{\mathrm{T}}}{y_k^{\mathrm{T}} s_k}\Big) = \frac{a_n}{\|\eta_k\|} = \frac{\|\eta_k\| a_n}{\|\eta_k\|^2} = \frac{y_k^{\mathrm{T}} s_k}{s_k^{\mathrm{T}} B_k s_k}\,,$$

and the identity above further implies that

$$\begin{aligned}\frac{y_k^{\mathrm{T}} s_k}{s_k^{\mathrm{T}} B_k s_k} &= \det\big(Q_k \Lambda_k^{-1/2} P_k^{\mathrm{T}} B_{k+1} P_k \Lambda_k^{-1/2} Q_k^{\mathrm{T}}\big) \\ &= \det(\Lambda_k^{-1/2}) \det(B_{k+1}) \det(\Lambda_k^{-1/2}) = \frac{\det(B_{k+1})}{\det(\Lambda_k)}\,.\end{aligned}$$

Therefore, the fact that $\det(\Lambda_k) = \det(B_k)$ shows that

$$\det(B_{k+1}) = \det(B_k) \frac{y_k^{\mathrm{T}} s_k}{s_k^{\mathrm{T}} B_k s_k}\,.$$

□

§6.4 Convergence Analysis

Proof (cont'd).

Note that $\eta_k = \Lambda_k^{1/2} P_k^{\mathrm{T}} s_k$ so that

$$\|\eta_k\|^2 = \eta_k^{\mathrm{T}} \eta_k = s_k^{\mathrm{T}} P_k \Lambda_k P_k^{\mathrm{T}} s_k = s_k^{\mathrm{T}} B_k s_k\,.$$

Using the properties of determinants,

$$\det\Big(\mathrm{I} - \mathrm{e}_n \mathrm{e}_n^{\mathrm{T}} + \frac{v_k v_k^{\mathrm{T}}}{y_k^{\mathrm{T}} s_k}\Big) = \frac{a_n}{\|\eta_k\|} = \frac{\|\eta_k\| a_n}{\|\eta_k\|^2} = \frac{y_k^{\mathrm{T}} s_k}{s_k^{\mathrm{T}} B_k s_k}\,,$$

and the identity above further implies that

$$\begin{aligned}\frac{y_k^{\mathrm{T}} s_k}{s_k^{\mathrm{T}} B_k s_k} &= \det\big(Q_k \Lambda_k^{-1/2} P_k^{\mathrm{T}} B_{k+1} P_k \Lambda_k^{-1/2} Q_k^{\mathrm{T}}\big)\\ &= \det(\Lambda_k^{-1/2})\det(B_{k+1})\det(\Lambda_k^{-1/2}) = \frac{\det(B_{k+1})}{\det(\Lambda_k)}\,.\end{aligned}$$

Therefore, the fact that $\det(\Lambda_k) = \det(B_k)$ shows that

$$\det(B_{k+1}) = \det(B_k)\frac{y_k^{\mathrm{T}} s_k}{s_k^{\mathrm{T}} B_k s_k}\,. \qquad \square$$

§6.4 Convergence Analysis

Proof (cont'd).

Define $m_k = \dfrac{y_k^{\mathrm{T}} s_k}{s_k^{\mathrm{T}} s_k}$, $M_k = \dfrac{y_k^{\mathrm{T}} y_k}{y_k^{\mathrm{T}} s_k}$, and $q_k = \dfrac{s_k^{\mathrm{T}} B_k s_k}{s_k^{\mathrm{T}} s_k}$. Then

$$\det(B_{k+1}) = \det(B_k) \frac{y_k^{\mathrm{T}} s_k}{s_k^{\mathrm{T}} s_k} \frac{s_k^{\mathrm{T}} s_k}{s_k^{\mathrm{T}} B_k s_k} = \det(B_k) \frac{m_k}{q_k}. \tag{45}$$

Moreover, since $s_k = \alpha_k p_k$,

$$\cos\theta_k = \frac{p_k^{\mathrm{T}} \nabla f_k}{\|p_k\| \|\nabla f_k\|} = \frac{p_k^{\mathrm{T}} B_k p_k}{\|p_k\| \|B_k p_k\|} = \frac{s_k^{\mathrm{T}} B_k s_k}{\|s_k\| \|B_k s_k\|}.$$

We then obtain that

$$\frac{\|B_k s_k\|^2}{s_k^{\mathrm{T}} B_k s_k} = \frac{\|B_k s_k\|^2 \|s_k\|^2}{(s_k^{\mathrm{T}} B_k s_k)^2} \frac{s_k^{\mathrm{T}} B_k s_k}{\|s_k\|^2} = \frac{q_k}{\cos^2\theta_k}$$

so that by taking the trace of B_{k+1} in the updating formula (22),

$$\operatorname{tr}(B_{k+1}) = \operatorname{tr}(B_k) - \frac{\|B_k s_k\|^2}{s_k^{\mathrm{T}} B_k s_k} + \frac{\|y_k\|^2}{y_k^{\mathrm{T}} s_k}. \tag{46}$$

$\square$

§6.4 Convergence Analysis

Proof (cont'd).

Define $m_k = \dfrac{y_k^{\mathrm{T}} s_k}{s_k^{\mathrm{T}} s_k}$, $M_k = \dfrac{y_k^{\mathrm{T}} y_k}{y_k^{\mathrm{T}} s_k}$, and $q_k = \dfrac{s_k^{\mathrm{T}} B_k s_k}{s_k^{\mathrm{T}} s_k}$. Then

$$\det(B_{k+1}) = \det(B_k) \frac{y_k^{\mathrm{T}} s_k}{s_k^{\mathrm{T}} s_k} \frac{s_k^{\mathrm{T}} s_k}{s_k^{\mathrm{T}} B_k s_k} = \det(B_k) \frac{m_k}{q_k} . \tag{45}$$

Moreover, since $s_k = \alpha_k p_k$,

$$\cos\theta_k = \frac{p_k^{\mathrm{T}} \nabla f_k}{\|p_k\| \|\nabla f_k\|} = \frac{p_k^{\mathrm{T}} B_k p_k}{\|p_k\| \|B_k p_k\|} = \frac{s_k^{\mathrm{T}} B_k s_k}{\|s_k\| \|B_k s_k\|} .$$

We then obtain that

$$\frac{\|B_k s_k\|^2}{s_k^{\mathrm{T}} B_k s_k} = \frac{\|B_k s_k\|^2 \|s_k\|^2}{(s_k^{\mathrm{T}} B_k s_k)^2} \frac{s_k^{\mathrm{T}} B_k s_k}{\|s_k\|^2} = \frac{q_k}{\cos^2\theta_k}$$

so that by taking the trace of B_{k+1} in the updating formula (22),

$$\operatorname{tr}(B_{k+1}) = \operatorname{tr}(B_k) - \frac{q_k}{\cos^2\theta_k} + M_k . \tag{46}$$

$\square$

§6.4 Convergence Analysis

Proof (cont'd).

Let $\psi : \mathrm{GL}(n, \mathbb{R}) \to \mathbb{R}$ be defined by

$$\psi(B) = \mathrm{tr}(B) - \ln|\det(B)|\,.$$

By the spectral decomposition of symmetric matrices and the inequality $x - 1 \geqslant \ln x$ for $x > 0$, we have

$$\psi(B) > 0 \quad \text{for all positive definite } B.$$

Using (45) and (46) we obtain

$$\begin{aligned}\psi(B_{k+1}) &= \mathrm{tr}(B_{k+1}) - \ln(\det(B_{k+1})) \\ &= \mathrm{tr}(B_k) - \frac{q_k}{\cos^2\theta_k} + M_k - \ln(\det(B_k)) - \ln m_k + \ln q_k \\ &= \psi(B_k) + \ln\cos^2\theta_k + (M_k - \ln m_k - 1) \\ &\quad + \Big[1 - \frac{q_k}{\cos^2\theta_k} + \ln\frac{q_k}{\cos^2\theta_k}\Big]\,. \end{aligned} \tag{47}$$

□

§6.4 Convergence Analysis

Proof (cont'd).

Let $\psi : \mathrm{GL}(n, \mathbb{R}) \to \mathbb{R}$ be defined by

$$\psi(B) = \mathrm{tr}(B) - \ln|\det(B)|\,.$$

By the spectral decomposition of symmetric matrices and the inequality $x - 1 \geqslant \ln x$ for $x > 0$, we have

$$\psi(B) > 0 \quad \text{for all positive definite } B.$$

Using (45) and (46) we obtain

$$\begin{aligned}\psi(B_{k+1}) &= \mathrm{tr}(B_{k+1}) - \ln(\det(B_{k+1})) \\ &= \mathrm{tr}(B_k) - \frac{q_k}{\cos^2\theta_k} + M_k - \ln(\det(B_k)) - \ln m_k + \ln q_k \\ &= \psi(B_k) + \ln\cos^2\theta_k + (M_k - \ln m_k - 1) \\ &\quad + \Big[1 - \frac{q_k}{\cos^2\theta_k} + \ln\frac{q_k}{\cos^2\theta_k}\Big]\,.\end{aligned} \tag{47}$$

□

§6.4 Convergence Analysis

Proof (cont'd).

Again by the inequality $x - 1 \geqslant \ln x$ for $x > 0$, the term inside the square brackets of (47) is non-positive so we have for all $k \in \mathbb{N}$,

$$\psi(B_{k+1}) \leqslant \psi(B_k) + (M_k - \ln m_k - 1) + \ln\cos^2\theta_k\,.$$

Therefore,

$$\sum_{j=0}^{k}\psi(B_{j+1}) \leqslant \sum_{j=0}^{k}\psi(B_j) + \sum_{j=0}^{k}(M_j - \ln m_j - 1) + \sum_{j=0}^{k}\ln\cos^2\theta_j$$

$$\Rightarrow \psi(B_{k+1}) \leqslant \psi(B_0) + \sum_{j=0}^{k}(M_j - \ln m_j - 1) + \sum_{j=0}^{k}\ln\cos^2\theta_j\,.$$

By (43) and (44), $m_k \geqslant m$ and $M_k \leqslant M$ for all $k \in \mathbb{N}$; thus

$$0 < \psi(B_{k+1}) \leqslant \psi(B_0) + c(k+1) + \sum_{j=0}^{k}\ln\cos^2\theta_j\,, \tag{48}$$

where $c = M - \ln m - 1$, W.L.O.G., is assumed to be positive. □

§6.4 Convergence Analysis

Proof (cont'd).

Again by the inequality $x - 1 \geqslant \ln x$ for $x > 0$, the term inside the square brackets of (47) is non-positive so we have for all $k \in \mathbb{N}$,

$$\psi(B_{k+1}) \leqslant \psi(B_k) + (M_k - \ln m_k - 1) + \ln \cos^2\theta_k \,.$$

Therefore,

$$\sum_{j=0}^{k} \psi(B_{j+1}) \leqslant \sum_{j=0}^{k} \psi(B_j) + \sum_{j=0}^{k} (M_j - \ln m_j - 1) + \sum_{j=0}^{k} \ln \cos^2\theta_j$$

$$\Rightarrow \psi(B_{k+1}) \leqslant \psi(B_0) + \sum_{j=0}^{k} (M_j - \ln m_j - 1) + \sum_{j=0}^{k} \ln \cos^2\theta_j \,.$$

By (43) and (44), $m_k \geqslant m$ and $M_k \leqslant M$ for all $k \in \mathbb{N}$; thus

$$0 < \psi(B_{k+1}) \leqslant \psi(B_0) + c(k+1) + \sum_{j=0}^{k} \ln \cos^2\theta_j \,, \tag{48}$$

where $c = M - \ln m - 1$, W.L.O.G., is assumed to be positive. □

§6.4 Convergence Analysis

Proof (cont'd).

Again by the inequality $x - 1 \geqslant \ln x$ for $x > 0$, the term inside the square brackets of (47) is non-positive so we have for all $k \in \mathbb{N}$,

$$\psi(B_{k+1}) \leqslant \psi(B_k) + (M_k - \ln m_k - 1) + \ln\cos^2\theta_k\,.$$

Therefore,

$$\begin{aligned}\sum_{j=0}^{k}\psi(B_{j+1}) &\leqslant \sum_{j=0}^{k}\psi(B_j) + \sum_{j=0}^{k}(M_j - \ln m_j - 1) + \sum_{j=0}^{k}\ln\cos^2\theta_j \\ \Rightarrow \psi(B_{k+1}) &\leqslant \psi(B_0) + \sum_{j=0}^{k}(M_j - \ln m_j - 1) + \sum_{j=0}^{k}\ln\cos^2\theta_j\,.\end{aligned}$$

By (43) and (44), $m_k \geqslant m$ and $M_k \leqslant M$ for all $k \in \mathbb{N}$; thus

$$0 < \psi(B_{k+1}) \leqslant \psi(B_0) + c(k+1) + \sum_{j=0}^{k}\ln\cos^2\theta_j\,, \tag{48}$$

where $c = M - \ln m - 1$, W.L.O.G., is assumed to be positive. □

§6.4 Convergence Analysis

Proof (cont'd).

Again by the inequality $x - 1 \geqslant \ln x$ for $x > 0$, the term inside the square brackets of (47) is non-positive so we have for all $k \in \mathbb{N}$,

$$\psi(B_{k+1}) \leqslant \psi(B_k) + (M_k - \ln m_k - 1) + \ln \cos^2\theta_k \,.$$

Therefore,

$$\sum_{j=0}^{k} \psi(B_{j+1}) \leqslant \sum_{j=0}^{k} \psi(B_j) + \sum_{j=0}^{k} (M_j - \ln m_j - 1) + \sum_{j=0}^{k} \ln \cos^2\theta_j$$

$$\Rightarrow \psi(B_{k+1}) \leqslant \psi(B_0) + \sum_{j=0}^{k} (M_j - \ln m_j - 1) + \sum_{j=0}^{k} \ln \cos^2\theta_j \,.$$

By (43) and (44), $m_k \geqslant m$ and $M_k \leqslant M$ for all $k \in \mathbb{N}$; thus

$$0 < \psi(B_{k+1}) \leqslant \psi(B_0) + c(k+1) + \sum_{j=0}^{k} \ln \cos^2 \theta_j \,, \tag{48}$$

where $c = M - \ln m - 1$, W.L.O.G., is assumed to be positive. □

§6.4 Convergence Analysis

Proof (cont'd).

Now we show that there exists $\delta > 0$ such that

$$\#\{\, j \in \mathbb{N} \,\big|\, |\cos\theta_j| \geqslant \delta \} = \infty \,.$$

Assume the contrary that $\cos\theta_j \to 0$. Then there exists $k_1 > 0$ such that

$$\ln\cos^2\theta_j < -2c \quad \text{for all } j > k_1,$$

where $c = M - \ln m - 1$ is the constant defined previously.

Using this inequality in (48) we find that for all $k > k_1$,

$$\begin{aligned} 0 &< \psi(B_0) + c(k+1) + \sum_{j=0}^{k_1} \ln\cos^2\theta_j + \sum_{j=k_1+1}^{k} (-2c) \\ &= \psi(B_0) + \sum_{j=0}^{k_1} \ln\cos^2\theta_j + 2ck_1 + c - ck\,, \end{aligned}$$

and the right-hand side approaches $-\infty$ as $k \to \infty$, a contradiction. □

§6.4 Convergence Analysis

Proof (cont'd).

Now we show that there exists $\delta > 0$ such that

$$\#\{\, j \in \mathbb{N} \,\big|\, |\cos\theta_j| \geqslant \delta \,\} = \infty\,.$$

Assume the contrary that $\cos\theta_j \to 0$. Then there exists $k_1 > 0$ such that

$$\ln\cos^2\theta_j < -2c \quad \text{for all } j > k_1,$$

where $c = M - \ln m - 1$ is the constant defined previously.

Using this inequality in (48) we find that for all $k > k_1$,

$$\begin{aligned} 0 &< \psi(B_0) + c(k+1) + \sum_{j=0}^{k_1} \ln\cos^2\theta_j + \sum_{j=k_1+1}^{k} (-2c) \\ &= \psi(B_0) + \sum_{j=0}^{k_1} \ln\cos^2\theta_j + 2ck_1 + c - ck\,, \end{aligned}$$

and the right-hand side approaches $-\infty$ as $k \to \infty$, a contradiction. □

§6.4 Convergence Analysis

Proof (cont'd).

Now we show that there exists $\delta > 0$ such that

$$\#\{ j \in \mathbb{N} \,\big|\, |\cos\theta_j| \geqslant \delta \} = \infty .$$

Assume the contrary that $\cos\theta_j \to 0$. Then there exists $k_1 > 0$ such that

$$\ln\cos^2\theta_j < -2c \quad \text{for all } j > k_1,$$

where $c = M - \ln m - 1$ is the constant defined previously.

Using this inequality in (48) we find that for all $k > k_1$,

$$\begin{aligned} 0 &< \psi(B_0) + c(k+1) + \sum_{j=0}^{k_1} \ln\cos^2\theta_j + \sum_{j=k_1+1}^{k} (-2c) \\ &= \psi(B_0) + \sum_{j=0}^{k_1} \ln\cos^2\theta_j + 2ck_1 + c - ck , \end{aligned}$$

and the right-hand side approaches $-\infty$ as $k \to \infty$, a contradiction. $\square$

§6.4 Convergence Analysis

Proof (cont'd).

Therefore, there exists a subsequence of indices $\{j_k\}_{k=1,2,\cdots}$ such that $\cos\theta_{j_k} \geqslant \delta > 0$. By Zoutendijk's result this limit implies that $\lim\limits_{k\to\infty} \|\nabla f_{j_k}\| = 0$, so we conclude that

$$\liminf_{k\to\infty} \|\nabla f_k\| = 0\,.$$

Finally we show that $x_\ell \to x_*$. Before proceeding, we show that $x_{j_k} \to x_*$. Nevertheless, by the mean value theorem,

$$(x_{j_k} - x_*)^{\mathrm{T}} \nabla f_{j_k} = (x_{j_k} - x_*)^{\mathrm{T}} (\nabla^2 f)(\tilde{x})(x_{j_k} - x_*)$$

for some $\tilde{x}$ on the line segment joining x_{j_k} and x_*. Since $\tilde{x} \in C$, by Assumption 6.1 and the Cauchy-Schwartz inequality we obtain

$$\begin{aligned} m\|x_{j_k} - x_*\|^2 &\leqslant (x_{j_k} - x_*)^{\mathrm{T}} (\nabla^2 f)(\tilde{x})(x_{j_k} - x_*) \\ &= (x_{j_k} - x_*)^{\mathrm{T}} \nabla f_{j_k} \leqslant \|x_{j_k} - x_*\| \|\nabla f_{j_k}\|\,. \end{aligned}$$

□

§6.4 Convergence Analysis

Proof (cont'd).

Therefore, there exists a subsequence of indices $\{j_k\}_{k=1,2,\cdots}$ such that $\cos\theta_{j_k} \geqslant \delta > 0$. By Zoutendijk's result this limit implies that $\lim\limits_{k\to\infty} \|\nabla f_{j_k}\| = 0$, so we conclude that

$$\liminf_{k\to\infty} \|\nabla f_k\| = 0\,.$$

Finally we show that $x_\ell \to x_*$. Before proceeding, we show that $x_{j_k} \to x_*$. Nevertheless, by the mean value theorem,

$$(x_{j_k} - x_*)^{\mathrm{T}} \nabla f_{j_k} = (x_{j_k} - x_*)^{\mathrm{T}} (\nabla^2 f)(\widetilde{x})(x_{j_k} - x_*)$$

for some $\widetilde{x}$ on the line segment joining x_{j_k} and x_*. Since $\widetilde{x} \in C$, by Assumption 6.1 and the Cauchy-Schwartz inequality we obtain

$$\begin{aligned} m\|x_{j_k} - x_*\|^2 &\leqslant (x_{j_k} - x_*)^{\mathrm{T}} (\nabla^2 f)(\widetilde{x})(x_{j_k} - x_*) \\ &= (x_{j_k} - x_*)^{\mathrm{T}} \nabla f_{j_k} \leqslant \|x_{j_k} - x_*\| \|\nabla f_{j_k}\|\,. \end{aligned}$$

□

§6.4 Convergence Analysis

Proof (cont'd).

Therefore, there exists a subsequence of indices $\{j_k\}_{k=1,2,\cdots}$ such that $\cos\theta_{j_k} \geqslant \delta > 0$. By Zoutendijk's result this limit implies that $\lim\limits_{k\to\infty} \|\nabla f_{j_k}\| = 0$, so we conclude that

$$\liminf_{k\to\infty} \|\nabla f_k\| = 0\,.$$

Finally we show that $x_\ell \to x_*$. Before proceeding, we show that $x_{j_k} \to x_*$. Nevertheless, by the mean value theorem,

$$(x_{j_k} - x_*)^{\mathrm{T}} \nabla f_{j_k} = (x_{j_k} - x_*)^{\mathrm{T}} (\nabla^2 f)(\widetilde{x})(x_{j_k} - x_*)$$

for some $\widetilde{x}$ on the line segment joining x_{j_k} and x_*. Since $\widetilde{x} \in C$, by Assumption 6.1 and the Cauchy-Schwartz inequality we obtain

$$\begin{aligned} m\|x_{j_k} - x_*\|^2 &\leqslant (x_{j_k} - x_*)^{\mathrm{T}} (\nabla^2 f)(\widetilde{x})(x_{j_k} - x_*) \\ &= (x_{j_k} - x_*)^{\mathrm{T}} \nabla f_{j_k} \leqslant \|x_{j_k} - x_*\| \|\nabla f_{j_k}\|\,. \end{aligned}$$

□

§6.4 Convergence Analysis

Proof (cont'd).

Since $\nabla f_{j_k} \to 0$, we conclude that $x_{j_k} \to x_*$.

By Taylor's Theorem, Assumption 6.1 implies that

$$f(x) \geqslant f(x_*) + \frac{m}{2}\|x - x_*\|^2 \qquad \forall\, x \in C;$$

thus

$$\|x_\ell - x_*\|^2 \leqslant \frac{2}{m}\big[f(x_\ell) - f(x_*)\big] \quad \forall\, \ell \in \mathbb{N}.$$

In particular, for all $k \in \mathbb{N}$ and $\ell > j_k$, we have

$$\|x_\ell - x_*\|^2 \leqslant \frac{2}{m}\big[f(x_\ell) - f(x_*)\big] \leqslant \frac{2}{m}\big[f(x_{j_k}) - f(x_*)\big].$$

Passing to the limit as $\ell \to \infty$, we obtain

$$\limsup_{\ell\to\infty} \|x_\ell - x_*\|^2 \leqslant \frac{2}{m}\big[f(x_{j_k}) - f(x_*)\big] \quad \forall\, k \in \mathbb{N}.$$

Since the right-hand side converges to 0 as $k \to \infty$, we conclude that $\limsup_{\ell\to\infty} \|x_\ell - x_*\| = 0$, establishing the result. □

§6.4 Convergence Analysis

Proof (cont'd).

Since $\nabla f_{j_k} \to 0$, we conclude that $x_{j_k} \to x_*$.

By Taylor's Theorem, Assumption 6.1 implies that

$$f(x) \geqslant f(x_*) + \frac{m}{2}\|x - x_*\|^2 \qquad \forall\, x \in C;$$

thus

$$\|x_\ell - x_*\|^2 \leqslant \frac{2}{m}\big[f(x_\ell) - f(x_*)\big] \quad \forall\, \ell \in \mathbb{N}.$$

In particular, for all $k \in \mathbb{N}$ and $\ell > j_k$, we have

$$\|x_\ell - x_*\|^2 \leqslant \frac{2}{m}\big[f(x_\ell) - f(x_*)\big] \leqslant \frac{2}{m}\big[f(x_{j_k}) - f(x_*)\big].$$

Passing to the limit as $\ell \to \infty$, we obtain

$$\limsup_{\ell\to\infty} \|x_\ell - x_*\|^2 \leqslant \frac{2}{m}\big[f(x_{j_k}) - f(x_*)\big] \quad \forall\, k \in \mathbb{N}.$$

Since the right-hand side converges to 0 as $k \to \infty$, we conclude that $\limsup_{\ell\to\infty} \|x_\ell - x_*\| = 0$, establishing the result. □

§6.4 Convergence Analysis

Proof (cont'd).

Since $\nabla f_{j_k} \to 0$, we conclude that $x_{j_k} \to x_*$.

By Taylor's Theorem, Assumption 6.1 implies that

$$f(x) \geqslant f(x_*) + \frac{m}{2}\|x - x_*\|^2 \qquad \forall\, x \in C;$$

thus

$$\|x_\ell - x_*\|^2 \leqslant \frac{2}{m}\big[f(x_\ell) - f(x_*)\big] \quad \forall\, \ell \in \mathbb{N}\,.$$

In particular, for all $k \in \mathbb{N}$ and $\ell > j_k$, we have

$$\|x_\ell - x_*\|^2 \leqslant \frac{2}{m}\big[f(x_\ell) - f(x_*)\big] \leqslant \frac{2}{m}\big[f(x_{j_k}) - f(x_*)\big]\,.$$

Passing to the limit as $\ell \to \infty$, we obtain

$$\limsup_{\ell \to \infty} \|x_\ell - x_*\|^2 \leqslant \frac{2}{m}\big[f(x_{j_k}) - f(x_*)\big] \quad \forall\, k \in \mathbb{N}\,.$$

Since the right-hand side converges to 0 as $k \to \infty$, we conclude that $\limsup_{\ell \to \infty} \|x_\ell - x_*\| = 0$, establishing the result. $\square$

§6.4 Convergence Analysis

The theorem above can be shown to hold for all $\phi_k \in [0,1)$ in

$$B_{k+1} = B_k - \frac{B_k s_k s_k^{\mathrm{T}} B_k}{s_k^{\mathrm{T}} B_k s_k} + \frac{y_k y_k^{\mathrm{T}}}{y_k^{\mathrm{T}} s_k} + \phi_k (s_k^{\mathrm{T}} B_k s_k) v_k v_k^{\mathrm{T}}\,, \tag{34}$$

but the argument seems to break down as $\phi_k \to 1^-$ because some of the self-correcting properties of the update are weakened considerably.

An extension of the analysis just given shows that the rate of convergence of the iterates is linear. In particular, we can show that the sequence $\|x_k - x_*\|$ converges to zero rapidly enough that

$$\sum_{k=1}^{\infty} \|x_k - x_*\| < \infty \tag{49}$$

We will not prove this claim, but rather establish that if (49) holds, then the rate of convergence is actually superlinear.

§6.4 Convergence Analysis

The theorem above can be shown to hold for all $\phi_k \in [0,1)$ in

$$B_{k+1} = B_k - \frac{B_k s_k s_k^{\mathrm{T}} B_k}{s_k^{\mathrm{T}} B_k s_k} + \frac{y_k y_k^{\mathrm{T}}}{y_k^{\mathrm{T}} s_k} + \phi_k (s_k^{\mathrm{T}} B_k s_k) v_k v_k^{\mathrm{T}}, \tag{34}$$

but the argument seems to break down as $\phi_k \to 1^-$ because some of the self-correcting properties of the update are weakened considerably.

An extension of the analysis just given shows that the rate of convergence of the iterates is linear. In particular, we can show that the sequence $\|x_k - x_*\|$ converges to zero rapidly enough that

$$\sum_{k=1}^{\infty} \|x_k - x_*\| < \infty \tag{49}$$

We will not prove this claim, but rather establish that if (49) holds, then the rate of convergence is actually superlinear.

§6.4 Convergence Analysis

- **Superlinear convergence of the BFGS method**

The analysis of this section makes use of the Dennis and Moré characterization

$$\lim_{k\to\infty} \frac{\|(B_k - \nabla^2 f(x_*))p_k\|}{\|p_k\|} = 0$$

of superlinear convergence. It applies to general nonlinear – not just convex – objective functions. For the results that follow we need to make an additional assumption.

Assumption 6.2.

The Hessian $\nabla^2 f$ is Lipschitz continuous at x_*; that is, there exist $L, \delta > 0$ such that

$$\|(\nabla^2 f)(x) - (\nabla^2 f)(x_*)\| \leqslant L\|x - x_*\| \quad \forall\, x \in B(x_*, \delta)\,.$$

§6.4 Convergence Analysis

- **Superlinear convergence of the BFGS method**

The analysis of this section makes use of the Dennis and Moré characterization

$$\lim_{k\to\infty} \frac{\|(B_k - \nabla^2 f(x_*))p_k\|}{\|p_k\|} = 0$$

of superlinear convergence. It applies to general nonlinear – not just convex – objective functions. For the results that follow we need to make an additional assumption.

Assumption 6.2.

The Hessian $\nabla^2 f$ is Lipschitz continuous at x_*; that is, there exist $L, \delta > 0$ such that

$$\|(\nabla^2 f)(x) - (\nabla^2 f)(x_*)\| \leqslant L\|x - x_*\| \quad \forall\, x \in B(x_*, \delta)\,.$$

§6.4 Convergence Analysis

Theorem

Suppose that f is twice continuously differentiable and that the iterates generated by the BFGS algorithm converge to a minimizer x_* at which $\nabla^2 f_*$ is positive definite and Assumption 6.2 holds. Suppose also that

$$\sum_{k=1}^{\infty} \|x_k - x_*\| < \infty \tag{49}$$

holds. Then x_k converges to x_* at a superlinear rate.

Proof.

We first show that Assumption 6.1 is satisfied near x_*. Since $\nabla^2 f_*$ is positive definite, by the continuity of $\nabla^2 f$ we find that there exists $\delta > 0$ such that

$$m\|z\|^2 \leqslant z^{\mathrm{T}}(\nabla^2 f)(x)z \leqslant M\|z\|^2 \quad \forall\, x \in B(x_*, \delta)\,. \qquad \square$$

§6.4 Convergence Analysis

Theorem

Suppose that f is twice continuously differentiable and that the iterates generated by the BFGS algorithm converge to a minimizer x_* at which $\nabla^2 f_*$ is positive definite and Assumption 6.2 holds. Suppose also that

$$\sum_{k=1}^{\infty} \|x_k - x_*\| < \infty \tag{49}$$

holds. Then x_k converges to x_* at a superlinear rate.

Proof.

We first show that Assumption 6.1 is satisfied near x_*. Since $\nabla^2 f_*$ is positive definite, by the continuity of $\nabla^2 f$ we find that there exists $\delta > 0$ such that

$$m\|z\|^2 \leqslant z^{\mathrm{T}}(\nabla^2 f)(x)z \leqslant M\|z\|^2 \quad \forall\, x \in B(x_*, \delta)\,. \qquad \square$$

§6.4 Convergence Analysis

Proof (cont'd).

Since $x_k \to x_*$, W.L.O.G. we can assume that

$$x_0 \in B(x_*, \delta) \quad \text{and} \quad f(x_0) - f(x_*) < \frac{m\delta^2}{8}\,.$$

Note that by Taylor's theorem, we have

$$f(x) \geqslant f(x_*) + \frac{m}{2}\|x - x_*\|^2 \quad \forall\, x \in B(x_*, \delta)\,.$$

Therefore, if $f(x) \leqslant f(x_0)$ and $x \in B(x_*, \delta)$, we have

$$\|x - x_*\| \leqslant \sqrt{\frac{2\big[f(x_0) - f(x_*)\big]}{m}} < \frac{\delta}{2}\,.$$

This shows that the level set $S = \big\{x \,\big|\, f(x) \leqslant f(x_0)\big\}$ has at least two connected components: one inside $B(x_*, \delta/2)$ and one outside $B(x_*, \delta)$. Since BFGS algorithm generates sequence of iterates whose function value decreases, W.L.O.G. we can assume that Assumption 6.1 is satisfied. □

§6.4 Convergence Analysis

Proof (cont'd).

Since $x_k \to x_*$, W.L.O.G. we can assume that

$$x_0 \in B(x_*, \delta) \quad \text{and} \quad f(x_0) - f(x_*) < \frac{m\delta^2}{8}.$$

Note that by Taylor's theorem, we have

$$f(x) \geqslant f(x_*) + \frac{m}{2}\|x - x_*\|^2 \quad \forall\, x \in B(x_*, \delta).$$

Therefore, if $f(x) \leqslant f(x_0)$ and $x \in B(x_*, \delta)$, we have

$$\|x - x_*\| \leqslant \sqrt{\frac{2\big[f(x_0) - f(x_*)\big]}{m}} < \frac{\delta}{2}.$$

This shows that the level set $S = \big\{x \,\big|\, f(x) \leqslant f(x_0)\big\}$ has at least two connected components: one inside $B(x_*, \delta/2)$ and one outside $B(x_*, \delta)$. Since BFGS algorithm generates sequence of iterates whose function value decreases, W.L.O.G. we can assume that Assumption 6.1 is satisfied. □

§6.4 Convergence Analysis

Proof (cont'd).

By the Dennis and Moré characterization, to show superlinear convergence of the BFGS algorithm we need to show that

$$\lim_{k\to\infty} \frac{\|(B_k - G_*)s_k\|}{\|s_k\|} = 0\,,$$

where we recall that $G_* = (\nabla^2 f)(x_*)$. By the boundedness and the positive definiteness of G_*, it is equivalent to that

$$\lim_{k\to\infty} \frac{\|G_*^{-1/2}(B_k - G_*)s_k\|}{\|G_*^{1/2}s_k\|} = 0\,. \tag{50}$$

Define the quantities

$$\widetilde{s}_k = G_*^{1/2}s_k\,, \quad \widetilde{y}_k = G_*^{-1/2}y_k\,, \quad \widetilde{B}_k = G_*^{-1/2}B_kG_*^{-1/2}\,.$$

It suffices to show that

$$\lim_{k\to\infty} \frac{\|(\widetilde{B}_k - \mathrm{I})\widetilde{s}_k\|}{\|\widetilde{s}_k\|} = 0\,. \tag{50'}$$

□

§6.4 Convergence Analysis

Proof (cont'd).

By the Dennis and Moré characterization, to show superlinear convergence of the BFGS algorithm we need to show that

$$\lim_{k\to\infty} \frac{\|(B_k - G_*)s_k\|}{\|s_k\|} = 0\,,$$

where we recall that $G_* = (\nabla^2 f)(x_*)$. By the boundedness and the positive definiteness of G_*, it is equivalent to that

$$\lim_{k\to\infty} \frac{\|G_*^{-1/2}(B_k - G_*)s_k\|}{\|G_*^{1/2}s_k\|} = 0\,. \tag{50}$$

Define the quantities

$$\widetilde{s}_k = G_*^{1/2}s_k\,, \quad \widetilde{y}_k = G_*^{-1/2}y_k\,, \quad \widetilde{B}_k = G_*^{-1/2}B_kG_*^{-1/2}\,.$$

It suffices to show that

$$\lim_{k\to\infty} \frac{\|(\widetilde{B}_k - \mathrm{I})\widetilde{s}_k\|}{\|\widetilde{s}_k\|} = 0\,. \tag{50'}$$

□

§6.4 Convergence Analysis

Proof (cont'd).

By pre- and post-multiplying the BFGS update formula (22) by $G_*^{-1/2}$ and grouping terms appropriately, we obtain

$$\widetilde{B}_{k+1} = \widetilde{B}_k - \frac{\widetilde{B}_k \widetilde{s}_k \widetilde{s}_k^{\mathrm{T}} \widetilde{B}_k}{\widetilde{s}_k^{\mathrm{T}} \widetilde{B}_k \widetilde{s}_k} + \frac{\widetilde{y}_k \widetilde{y}_k^{\mathrm{T}}}{\widetilde{y}_k^{\mathrm{T}} \widetilde{s}_k}\,. \tag{22'}$$

Since this expression has precisely the same form as the BFGS formula (22) and Assumption 6.1 is satisfied (near x_*), it follows from the argument leading to (47) that

$$\begin{aligned}\psi(\widetilde{B}_{k+1}) = \psi(\widetilde{B}_k) + \ln\cos^2\widetilde{\theta}_k + (\widetilde{M}_k - \ln\widetilde{m}_k - 1) \\ + \left[1 - \frac{\widetilde{q}_k}{\cos^2\widetilde{\theta}_k} + \ln\frac{\widetilde{q}_k}{\cos^2\widetilde{\theta}_k}\right],\end{aligned} \tag{51}$$

where

$$\cos\widetilde{\theta}_k = \frac{\widetilde{s}_k^{\mathrm{T}} \widetilde{B}_k \widetilde{s}_k}{\|\widetilde{s}_k\| \|\widetilde{B}_k \widetilde{s}_k\|}, \quad \widetilde{q}_k = \frac{\widetilde{s}_k^{\mathrm{T}} \widetilde{B}_k \widetilde{s}_k}{\|\widetilde{s}_k\|^2}, \quad \widetilde{M}_k = \frac{\widetilde{y}_k^{\mathrm{T}} \widetilde{y}_k}{\widetilde{y}_k^{\mathrm{T}} \widetilde{s}_k}, \quad \widetilde{m}_k = \frac{\widetilde{y}_k^{\mathrm{T}} \widetilde{s}_k}{\widetilde{s}_k^{\mathrm{T}} \widetilde{s}_k}\,. \quad \square$$

§6.4 Convergence Analysis

Proof (cont'd).

Next we show that

$$\frac{\|\tilde{y}_k - \tilde{s}_k\|}{\|\tilde{s}_k\|} \leqslant \bar{c}\left[\|x_{k+1} - x_*\| + \|x_k - x_*\|\right] \tag{52}$$

for some constant $\bar{c}$. By Assumption 6.2, and recalling the definition

$$\bar{G}_k = \left[\int_0^1 (\nabla^2 f)(x_k + \tau\alpha_k p_k)\, d\tau\right], \tag{11}$$

we have

$$\begin{aligned}
\|\bar{G}_k - G_*\| &\leqslant \int_0^1 \|(\nabla^2 f)(x_k + \tau\alpha_k p_k) - (\nabla^2 f)(x_*)\|\, d\tau \\
&\leqslant \int_0^1 L\|x_k + \tau\alpha_k p_k - x_*\|\, d\tau \\
&\leqslant L\int_0^1 \|\tau(x_{k+1} - x_*) + (1-\tau)(x_k - x_*)\|\, d\tau \\
&\leqslant \frac{L}{2}\left[\|x_{k+1} - x_*\| + \|x_k - x_*\|\right].
\end{aligned}$$

□

§6.4 Convergence Analysis

Proof (cont'd).

Next we show that

$$\frac{\|\widetilde{y}_k - \widetilde{s}_k\|}{\|\widetilde{s}_k\|} \leqslant \bar{c}\left[\|x_{k+1} - x_*\| + \|x_k - x_*\|\right] \tag{52}$$

for some constant $\bar{c}$. By Assumption 6.2, and recalling the definition

$$\bar{G}_k = \left[\int_0^1 (\nabla^2 f)(x_k + \tau\alpha_k p_k)\, d\tau\right], \tag{11}$$

we have

$$\begin{aligned}
\|\bar{G}_k - G_*\| &\leqslant \int_0^1 \|(\nabla^2 f)(x_k + \tau\alpha_k p_k) - (\nabla^2 f)(x_*)\|\, d\tau \\
&\leqslant \int_0^1 L\|x_k + \tau\alpha_k p_k - x_*\|\, d\tau \\
&\leqslant L\int_0^1 \|\tau(x_{k+1} - x_*) + (1-\tau)(x_k - x_*)\|\, d\tau \\
&\leqslant \frac{L}{2}\left[\|x_{k+1} - x_*\| + \|x_k - x_*\|\right].
\end{aligned}$$

□

§6.4 Convergence Analysis

Proof (cont'd).

Recalling the identity $y_k = \bar{G}_k s_k$ (12), we have

$$y_k - G_* s_k = (\bar{G}_k - G_*) s_k \, ;$$

thus

$$\widetilde{y}_k - \widetilde{s}_k = G_*^{-1/2} (\bar{G}_k - G_*) G_*^{-1/2} \widetilde{s}_k \, .$$

Using the estimate for $\|\bar{G}_k - G_*\|$ from the previous page, we obtain

$$\begin{aligned} \|\widetilde{y}_k - \widetilde{s}_k\| &\leqslant \|G_*^{-1/2}\|^2 \|\widetilde{s}_k\| \|\bar{G}_k - G_*\| \\ &\leqslant \frac{1}{2} \|G_*^{-1/2}\|^2 \|\widetilde{s}_k\| \, L \left[\|x_{k+1} - x_*\| + \|x_k - x_*\| \right] , \end{aligned}$$

so, by setting $\bar{c} = \dfrac{1}{2} \|G_*^{-1/2}\|^2 L$, we conclude

$$\frac{\|\widetilde{y}_k - \widetilde{s}_k\|}{\|\widetilde{s}_k\|} \leqslant \bar{c} \left[\|x_{k+1} - x_*\| + \|x_k - x_*\| \right] . \tag{52}$$

□

§6.4 Convergence Analysis

Proof (cont'd).

Let $\varepsilon_k = \|x_{k+1} - x_*\| + \|x_k - x_*\|$. From (52),

$$\|\widetilde{y}_k\| - \|\widetilde{s}_k\| \leqslant \bar{c}\,\varepsilon_k \|\widetilde{s}_k\|, \quad \|\widetilde{s}_k\| - \|\widetilde{y}_k\| \leqslant \bar{c}\,\varepsilon_k \|\widetilde{s}_k\|,$$

so that

$$(1 - \bar{c}\,\varepsilon_k)\|\widetilde{s}_k\| \leqslant \|\widetilde{y}_k\| \leqslant (1 + \bar{c}\,\varepsilon_k)\|\widetilde{s}_k\| . \tag{53}$$

By squaring (52) and using (53), we obtain

$$\begin{aligned}(1 - \bar{c}\,\varepsilon_k)^2\|\widetilde{s}_k\|^2 - 2\widetilde{y}_k^{\mathrm{T}}\widetilde{s}_k + \|\widetilde{s}_k\|^2 &\leqslant \|\widetilde{y}_k\|^2 - 2\widetilde{y}_k^{\mathrm{T}}\widetilde{s}_k + \|\widetilde{s}_k\|^2 \\ &\leqslant \bar{c}^2\varepsilon_k^2\|\widetilde{s}_k\|^2,\end{aligned}$$

and therefore

$$2\widetilde{y}_k^{\mathrm{T}}\widetilde{s}_k \geqslant (1 - 2\bar{c}\,\varepsilon_k + \bar{c}^2\varepsilon_k^2 + 1 - \bar{c}^2\varepsilon_k^2)\|\widetilde{s}_k\|^2 = 2(1 - \bar{c}\,\varepsilon_k)\|\widetilde{s}_k\|^2.$$

It follows from the definition of $\widetilde{m}_k$ that

$$\widetilde{m}_k = \frac{\widetilde{y}_k^{\mathrm{T}}\widetilde{s}_k}{\widetilde{s}_k^{\mathrm{T}}\widetilde{s}_k} \geqslant 1 - \bar{c}\,\varepsilon_k . \tag{54}$$

□

§6.4 Convergence Analysis

Proof (cont'd).

Let $\varepsilon_k = \|x_{k+1} - x_*\| + \|x_k - x_*\|$. From (52),

$$\|\widetilde{y}_k\| - \|\widetilde{s}_k\| \leqslant \bar{c}\,\varepsilon_k\|\widetilde{s}_k\|, \quad \|\widetilde{s}_k\| - \|\widetilde{y}_k\| \leqslant \bar{c}\,\varepsilon_k\|\widetilde{s}_k\|,$$

so that

$$(1 - \bar{c}\,\varepsilon_k)\|\widetilde{s}_k\| \leqslant \|\widetilde{y}_k\| \leqslant (1 + \bar{c}\,\varepsilon_k)\|\widetilde{s}_k\| . \tag{53}$$

By squaring (52) and using (53), we obtain

$$\begin{aligned}(1 - \bar{c}\,\varepsilon_k)^2\|\widetilde{s}_k\|^2 - 2\widetilde{y}_k^{\mathrm{T}}\widetilde{s}_k + \|\widetilde{s}_k\|^2 &\leqslant \|\widetilde{y}_k\|^2 - 2\widetilde{y}_k^{\mathrm{T}}\widetilde{s}_k + \|\widetilde{s}_k\|^2 \\ &\leqslant \bar{c}^2\varepsilon_k^2\|\widetilde{s}_k\|^2,\end{aligned}$$

and therefore

$$2\widetilde{y}_k^{\mathrm{T}}\widetilde{s}_k \geqslant (1 - 2\bar{c}\,\varepsilon_k + \bar{c}^2\varepsilon_k^2 + 1 - \bar{c}^2\varepsilon_k^2)\|\widetilde{s}_k\|^2 = 2(1 - \bar{c}\,\varepsilon_k)\|\widetilde{s}_k\|^2.$$

It follows from the definition of $\widetilde{m}_k$ that

$$\widetilde{m}_k = \frac{\widetilde{y}_k^{\mathrm{T}}\widetilde{s}_k}{\widetilde{s}_k^{\mathrm{T}}\widetilde{s}_k} \geqslant 1 - \bar{c}\,\varepsilon_k . \tag{54}$$

□

§6.4 Convergence Analysis

Proof (cont'd).

Let $\varepsilon_k = \|x_{k+1} - x_*\| + \|x_k - x_*\|$. From (52),

$$\|\widetilde{y}_k\| - \|\widetilde{s}_k\| \leqslant \bar{c}\,\varepsilon_k \|\widetilde{s}_k\|, \quad \|\widetilde{s}_k\| - \|\widetilde{y}_k\| \leqslant \bar{c}\,\varepsilon_k \|\widetilde{s}_k\|,$$

so that

$$(1 - \bar{c}\,\varepsilon_k)\|\widetilde{s}_k\| \leqslant \|\widetilde{y}_k\| \leqslant (1 + \bar{c}\,\varepsilon_k)\|\widetilde{s}_k\|. \tag{53}$$

By squaring (52) and using (53), we obtain

$$\begin{aligned}(1 - \bar{c}\,\varepsilon_k)^2\|\widetilde{s}_k\|^2 - 2\widetilde{y}_k^{\mathrm{T}}\widetilde{s}_k + \|\widetilde{s}_k\|^2 &\leqslant \|\widetilde{y}_k\|^2 - 2\widetilde{y}_k^{\mathrm{T}}\widetilde{s}_k + \|\widetilde{s}_k\|^2 \\ &\leqslant \bar{c}^2\varepsilon_k^2\|\widetilde{s}_k\|^2,\end{aligned}$$

and therefore

$$2\widetilde{y}_k^{\mathrm{T}}\widetilde{s}_k \geqslant (1 - 2\bar{c}\,\varepsilon_k + \bar{c}^2\varepsilon_k^2 + 1 - \bar{c}^2\varepsilon_k^2)\|\widetilde{s}_k\|^2 = 2(1 - \bar{c}\,\varepsilon_k)\|\widetilde{s}_k\|^2.$$

It follows from the definition of $\widetilde{m}_k$ that

$$\widetilde{m}_k = \frac{\widetilde{y}_k^{\mathrm{T}}\widetilde{s}_k}{\widetilde{s}_k^{\mathrm{T}}\widetilde{s}_k} \geqslant 1 - \bar{c}\,\varepsilon_k. \tag{54}$$

□

§6.4 Convergence Analysis

Proof (cont'd).

By combining (53) and (54), we obtain also that

$$\widetilde{M}_k = \frac{\widetilde{y}_k^{\mathrm{T}} \widetilde{y}_k}{\widetilde{y}_k^{\mathrm{T}} \widetilde{s}_k} \leqslant \frac{(1+\bar{c}\,\varepsilon_k)^2}{1-\bar{c}\,\varepsilon_k}\,. \tag{55}$$

Since $x_k \to x_*$, we have that $\varepsilon_k \to 0$; thus there exists $K > 0$ such that $\bar{c}\,\varepsilon_k < \dfrac{1}{2}$ for all $k \geqslant K$. Using (55) we find that

$$\widetilde{M}_k \leqslant 1 + \frac{7\bar{c}/2}{1-\bar{c}\,\varepsilon_k}\varepsilon_k \leqslant 1 + 7\bar{c}\,\varepsilon_k \equiv 1 + c\,\varepsilon_k \quad \forall\, k \geqslant K\,. \tag{56}$$

Again by the non-positiveness of the function $h(t) = 1 - t + \ln t$, we conclude that

$$\frac{-x}{1-x} - \ln(1-x) = h\Big(\frac{1}{1-x}\Big) \leqslant 0 \quad \forall\, x < 1\,.$$

Therefore,

$$\ln(1-\bar{c}\,\varepsilon_k) \geqslant \frac{-\bar{c}\,\varepsilon_k}{1-\bar{c}\,\varepsilon_k} \geqslant -2\bar{c}\,\varepsilon_k \quad \forall\, k \geqslant K\,.$$

□

§6.4 Convergence Analysis

Proof (cont'd).

By combining (53) and (54), we obtain also that

$$\widetilde{M}_k = \frac{\widetilde{y}_k^{\mathrm{T}} \widetilde{y}_k}{\widetilde{y}_k^{\mathrm{T}} \widetilde{s}_k} \leqslant \frac{(1+\bar{c}\varepsilon_k)^2}{1-\bar{c}\varepsilon_k}\,. \tag{55}$$

Since $x_k \to x_*$, we have that $\varepsilon_k \to 0$; thus there exists $K > 0$ such that $\bar{c}\varepsilon_k < \dfrac{1}{2}$ for all $k \geqslant K$. Using (55) we find that

$$\widetilde{M}_k \leqslant 1 + \frac{7\bar{c}/2}{1-\bar{c}\varepsilon_k}\varepsilon_k \leqslant 1 + 7\bar{c}\varepsilon_k \equiv 1 + c\varepsilon_k \quad \forall\, k \geqslant K\,. \tag{56}$$

Again by the non-positiveness of the function $h(t) = 1 - t + \ln t$, we conclude that

$$\frac{-x}{1-x} - \ln(1-x) = h\Big(\frac{1}{1-x}\Big) \leqslant 0 \quad \forall\, x < 1\,.$$

Therefore,

$$\ln(1-\bar{c}\varepsilon_k) \geqslant \frac{-\bar{c}\varepsilon_k}{1-\bar{c}\varepsilon_k} \geqslant -2\bar{c}\varepsilon_k \quad \forall\, k \geqslant K\,.$$

□

§6.4 Convergence Analysis

Proof (cont'd).

By combining (53) and (54), we obtain also that

$$\widetilde{M}_k = \frac{\widetilde{y}_k^{\mathrm{T}} \widetilde{y}_k}{\widetilde{y}_k^{\mathrm{T}} \widetilde{s}_k} \leqslant \frac{(1+\bar{c}\varepsilon_k)^2}{1-\bar{c}\varepsilon_k}. \tag{55}$$

Since $x_k \to x_*$, we have that $\varepsilon_k \to 0$; thus there exists $K > 0$ such that $\bar{c}\varepsilon_k < \frac{1}{2}$ for all $k \geqslant K$. Using (55) we find that

$$\widetilde{M}_k \leqslant 1 + \frac{7\bar{c}/2}{1-\bar{c}\varepsilon_k}\varepsilon_k \leqslant 1 + 7\bar{c}\varepsilon_k \equiv 1 + c\varepsilon_k \quad \forall\, k \geqslant K. \tag{56}$$

Again by the non-positiveness of the function $h(t) = 1 - t + \ln t$, we conclude that

$$\frac{-x}{1-x} - \ln(1-x) = h\Big(\frac{1}{1-x}\Big) \leqslant 0 \quad \forall\, x < 1.$$

Therefore,

$$\ln(1-\bar{c}\varepsilon_k) \geqslant \frac{-\bar{c}\varepsilon_k}{1-\bar{c}\varepsilon_k} \geqslant -2\bar{c}\varepsilon_k \quad \forall\, k \geqslant K.$$

□

§6.4 Convergence Analysis

Proof (cont'd).

The inequality $\ln(1-\bar{c}\varepsilon_k) \geqslant -2\bar{c}\varepsilon_k$ for $k \geqslant K$ and (54) imply that

$$\ln \widetilde{m}_k \geqslant \ln(1-\bar{c}\varepsilon_k) \geqslant -2\bar{c}\varepsilon_k > -2c\varepsilon_k \quad \forall\, k \geqslant K. \tag{57}$$

We can now use (57) and the inequality

$$\widetilde{M}_k \leqslant 1 + c\varepsilon_k \quad \forall\, k \geqslant K \tag{56}$$

in the inequality

$$\begin{aligned}\psi(\widetilde{B}_{k+1}) &= \psi(\widetilde{B}_k) + \ln\cos^2\widetilde{\theta}_k + (\widetilde{M}_k - \ln\widetilde{m}_k - 1)\\ &\quad + \Big[1 - \frac{\widetilde{q}_k}{\cos^2\widetilde{\theta}_k} + \ln\frac{\widetilde{q}_k}{\cos^2\widetilde{\theta}_k}\Big]\end{aligned} \tag{51}$$

to obtain that

$$\begin{aligned}0 < \psi(\widetilde{B}_{k+1}) &\leqslant \psi(\widetilde{B}_k) + 3c\varepsilon_k + \ln\cos^2\widetilde{\theta}_k\\ &\quad + \Big[1 - \frac{\widetilde{q}_k}{\cos^2\widetilde{\theta}_k} + \ln\frac{\widetilde{q}_k}{\cos^2\widetilde{\theta}_k}\Big]\end{aligned} \quad \forall\, k \geqslant K. \tag{58}$$

§6.4 Convergence Analysis

Proof (cont'd).

The inequality $\ln(1-\bar{c}\varepsilon_k) \geqslant -2\bar{c}\varepsilon_k$ for $k \geqslant K$ and (54) imply that

$$\ln \widetilde{m}_k \geqslant \ln(1-\bar{c}\varepsilon_k) \geqslant -2\bar{c}\varepsilon_k > -2c\varepsilon_k \quad \forall\, k \geqslant K. \tag{57}$$

We can now use (57) and the inequality

$$\widetilde{M}_k \leqslant 1 + c\varepsilon_k \quad \forall\, k \geqslant K \tag{56}$$

in the inequality

$$\begin{aligned}\psi(\widetilde{B}_{k+1}) = \psi(\widetilde{B}_k) + \ln\cos^2\widetilde{\theta}_k + (\widetilde{M}_k - \ln\widetilde{m}_k - 1) \\ + \Big[1 - \frac{\widetilde{q}_k}{\cos^2\widetilde{\theta}_k} + \ln\frac{\widetilde{q}_k}{\cos^2\widetilde{\theta}_k}\Big]\end{aligned} \tag{51}$$

to obtain that

$$\begin{aligned}0 < \psi(\widetilde{B}_{k+1}) \leqslant \psi(\widetilde{B}_k) + 3c\varepsilon_k + \ln\cos^2\widetilde{\theta}_k \\ + \Big[1 - \frac{\widetilde{q}_k}{\cos^2\widetilde{\theta}_k} + \ln\frac{\widetilde{q}_k}{\cos^2\widetilde{\theta}_k}\Big]\end{aligned} \quad \forall\, k \geqslant K. \tag{58}$$

□

§6.4 Convergence Analysis

Proof (cont'd).

Rearranging terms in (58), by the non-positiveness of $\ln\cos^2\theta$ and the function $h(t) = 1 - t + \ln t$ we have

$$0 < \left[\ln\frac{1}{\cos^2\widetilde{\theta}_j} - \left(1 - \frac{\tilde{q}_j}{\cos^2\widetilde{\theta}_j} + \ln\frac{\tilde{q}_j}{\cos^2\widetilde{\theta}_j}\right)\right] \qquad \forall\, j \geqslant K\,.$$

$$\leqslant \left[\psi(\widetilde{B}_j) - \psi(\widetilde{B}_{j+1})\right] + 3c\,\varepsilon_j$$

By summing this expression, by the fact that $\psi(B) > 0$ for positive definite B we have that for $J > K$,

$$\sum_{j=K}^{J}\left(\ln\frac{1}{\cos^2\widetilde{\theta}_j} + \left|1 - \frac{\tilde{q}_j}{\cos^2\widetilde{\theta}_j} + \ln\frac{\tilde{q}_j}{\cos^2\widetilde{\theta}_j}\right|\right)$$

$$\leqslant \psi(\widetilde{B}_K) - \psi(\widetilde{B}_{J+1}) + 3c\sum_{j=K}^{J}\varepsilon_j$$

$$\leqslant \psi(\widetilde{B}_K) + 3c\sum_{j=K}^{J}\varepsilon_j\,.$$

□

§6.4 Convergence Analysis

Proof (cont'd).

Rearranging terms in (58), by the non-positiveness of $\ln\cos^2\theta$ and the function $h(t) = 1 - t + \ln t$ we have

$$\begin{aligned} 0 &< \Big[\ln\frac{1}{\cos^2\widetilde{\theta}_j} - \Big(1 - \frac{\widetilde{q}_j}{\cos^2\widetilde{\theta}_j} + \ln\frac{\widetilde{q}_j}{\cos^2\widetilde{\theta}_j}\Big)\Big] \\ &\leqslant \big[\psi(\widetilde{B}_j) - \psi(\widetilde{B}_{j+1})\big] + 3c\,\varepsilon_j \end{aligned} \qquad \forall\, j \geqslant K\,.$$

By summing this expression, by the fact that $\psi(B) > 0$ for positive definite B we have that for $J > K$,

$$\begin{aligned} \sum_{j=K}^{J}\Big(\ln\frac{1}{\cos^2\widetilde{\theta}_j} + \Big|1 - \frac{\widetilde{q}_j}{\cos^2\widetilde{\theta}_j} + \ln\frac{\widetilde{q}_j}{\cos^2\widetilde{\theta}_j}\Big|\Big) &\leqslant \psi(\widetilde{B}_K) - \psi(\widetilde{B}_{J+1}) + 3c\sum_{j=K}^{J}\varepsilon_j \\ &\leqslant \psi(\widetilde{B}_K) + 3c\sum_{j=K}^{J}\varepsilon_j\,. \end{aligned}$$

□

§6.4 Convergence Analysis

Proof (cont'd).

Making use of the condition $\sum_{k=1}^{\infty} \|x_k - x_*\| < \infty$ (49) we find that

$$\sum_{j=K}^{\infty} \varepsilon_j = \sum_{j=K}^{\infty} \big[\|x_{j+1} - x_*\| + \|x_j - x_*\|\big] \leqslant 2 \sum_{j=1}^{\infty} \|x_j - x_*\| < \infty .$$

Passing to the limit as $J \to \infty$, we conclude that

$$\sum_{j=K}^{\infty} \left(\ln \frac{1}{\cos^2 \tilde{\theta}_j} + \left| 1 - \frac{\tilde{q}_j}{\cos^2 \tilde{\theta}_j} + \ln \frac{\tilde{q}_j}{\cos^2 \tilde{\theta}_j} \right| \right) < \infty .$$

Since the term in the parenthesis is non-negative, we obtain the following two limits

$$\lim_{j\to\infty} \ln \frac{1}{\cos^2 \tilde{\theta}_j} = 0 , \qquad \lim_{j\to\infty} \left[1 - \frac{\tilde{q}_j}{\cos^2 \tilde{\theta}_j} + \ln \frac{\tilde{q}_j}{\cos^2 \tilde{\theta}_j} \right] = 0 ,$$

which further imply that

$$\lim_{j\to\infty} \cos \tilde{\theta}_j = 1, \qquad \lim_{j\to\infty} \tilde{q}_j = 1 . \qquad (59)$$

□

§6.4 Convergence Analysis

Proof (cont'd).

Making use of the condition $\sum\limits_{k=1}^{\infty} \|x_k - x_*\| < \infty$ (49) we find that

$$\sum_{j=K}^{\infty} \varepsilon_j = \sum_{j=K}^{\infty} \big[\|x_{j+1} - x_*\| + \|x_j - x_*\|\big] \leqslant 2 \sum_{j=1}^{\infty} \|x_j - x_*\| < \infty .$$

Passing to the limit as $J \to \infty$, we conclude that

$$\sum_{j=K}^{\infty} \left(\ln \frac{1}{\cos^2 \widetilde{\theta}_j} + \left| 1 - \frac{\widetilde{q}_j}{\cos^2 \widetilde{\theta}_j} + \ln \frac{\widetilde{q}_j}{\cos^2 \widetilde{\theta}_j} \right| \right) < \infty .$$

Since the term in the parenthesis is non-negative, we obtain the following two limits

$$\lim_{j\to\infty} \ln \frac{1}{\cos^2 \widetilde{\theta}_j} = 0 , \quad \lim_{j\to\infty} \left[1 - \frac{\widetilde{q}_j}{\cos^2 \widetilde{\theta}_j} + \ln \frac{\widetilde{q}_j}{\cos^2 \widetilde{\theta}_j} \right] = 0 ,$$

which further imply that

$$\lim_{j\to\infty} \cos \widetilde{\theta}_j = 1, \quad \lim_{j\to\infty} \widetilde{q}_j = 1 . \tag{59}$$

□

§6.4 Convergence Analysis

Proof (cont'd).

Finally, recalling the definition of $\cos\widetilde{\theta}_k$ and $\widetilde{q}_k$, we have

$$\frac{\|(\widetilde{B}_k - \mathrm{I})\widetilde{s}_k\|^2}{\|\widetilde{s}_k\|^2} = \frac{\|\widetilde{B}_k\widetilde{s}_k\|^2 - 2\widetilde{s}_k^{\mathrm{T}}\widetilde{B}_k\widetilde{s}_k + \widetilde{s}_k^{\mathrm{T}}\widetilde{s}_k}{\widetilde{s}_k^{\mathrm{T}}\widetilde{s}_k} = \frac{\widetilde{q}_k^2}{\cos^2\widetilde{\theta}_k} - 2\widetilde{q}_k + 1\,,$$

and the right-hand side converges to 0 because of (59); thus

$$\lim_{k\to\infty}\frac{\|(\widetilde{B}_k - \mathrm{I})\widetilde{s}_k\|}{\|\widetilde{s}_k\|} = 0 \tag{50'}$$

We remind the reader that (50') is equivalent to the Dennis-Moré characterization

$$\lim_{k\to\infty}\frac{\|(B_k - G_*)s_k\|}{\|s_k\|} = 0$$

of the superlinear convergence. Therefore, $x_k \to x_*$ at a superlinear rate. □

§6.4 Convergence Analysis

Proof (cont'd).

Finally, recalling the definition of $\cos\widetilde{\theta}_k$ and $\widetilde{q}_k$, we have

$$\frac{\|(\widetilde{B}_k - \mathrm{I})\widetilde{s}_k\|^2}{\|\widetilde{s}_k\|^2} = \frac{\|\widetilde{B}_k\widetilde{s}_k\|^2 - 2\widetilde{s}_k^{\mathrm{T}}\widetilde{B}_k\widetilde{s}_k + \widetilde{s}_k^{\mathrm{T}}\widetilde{s}_k}{\widetilde{s}_k^{\mathrm{T}}\widetilde{s}_k} = \frac{\widetilde{q}_k^2}{\cos^2\widetilde{\theta}_k} - 2\widetilde{q}_k + 1\,,$$

and the right-hand side converges to 0 because of (59); thus

$$\lim_{k\to\infty}\frac{\|(\widetilde{B}_k - \mathrm{I})\widetilde{s}_k\|}{\|\widetilde{s}_k\|} = 0 \tag{50'}$$

We remind the reader that $(50')$ is equivalent to the Dennis-Moré characterization

$$\lim_{k\to\infty}\frac{\|(B_k - G_*)s_k\|}{\|s_k\|} = 0$$

of the superlinear convergence. Therefore, $x_k \to x_*$ at a superlinear rate. □

§6.4 Convergence Analysis

- **Convergence analysis of the SR1 method**

The convergence properties of the SR1 method are not as well understood as those of the BFGS method. **No** global results or local superlinear results like the previous two theorems have been established, except the results for quadratic functions discussed earlier. There is, however, an interesting result for the trust-region SR1 algorithm, Algorithm 6.2. It states that when the objective function has a unique stationary point and the condition

$$|s_k^{\mathrm{T}}(y_k - B_k s_k)| \geqslant r\|s_k\|\|y_k - B_k s_k\| \tag{29}$$

holds at every step (so that the SR1 update is never skipped) and the Hessian approximations B_k are uniformly bounded, then the iterates converge to x_* at an $(n+1)$-step superlinear rate. The result does not require exact solution of the trust-region sub-problem (30).

§6.4 Convergence Analysis

- **Convergence analysis of the SR1 method**

The convergence properties of the SR1 method are not as well understood as those of the BFGS method. **No** global results or local superlinear results like the previous two theorems have been established, except the results for quadratic functions discussed earlier. There is, however, an interesting result for the trust-region SR1 algorithm, Algorithm 6.2. It states that when the objective function has a unique stationary point and the condition

$$|s_k^{\mathrm{T}}(y_k - B_k s_k)| \geqslant r\|s_k\|\|y_k - B_k s_k\| \tag{29}$$

holds at every step (so that the SR1 update is never skipped) and the Hessian approximations B_k are uniformly bounded, then the iterates converge to x_* at an $(n+1)$-step superlinear rate. The result does not require exact solution of the trust-region sub-problem (30).

§6.4 Convergence Analysis

Theorem

Suppose that the iterates x_k are generated by Algorithm 6.2. Suppose also that the following conditions hold:

1. *The sequence of iterates does not terminate, but remains in a closed, bounded, convex set D, on which the function f is twice continuously differentiable, and in which f has a unique stationary point x_*;*
2. *the Hessian $\nabla^2 f(x_*)$ is positive definite, and $\nabla^2 f$ is Lipschitz continuous in a neighborhood of x_*;*
3. *the sequence of matrices $\{B_k\}$ is uniformly bounded;*
4. *condition (29) holds at every iteration, where r is some constant in $(0,1)$.*

Then $\lim\limits_{k\to\infty} x_k = x_*$, and we have that $\lim\limits_{k\to\infty} \dfrac{\|x_{k+n+1} - x_*\|}{\|x_k - x_*\|} = 0.$

§6.4 Convergence Analysis

Note that the BFGS method does not require the boundedness assumption ③ to hold. As we have mentioned already, the SR1 update does not necessarily maintain positive definiteness of the Hessian approximations B_k. In practice, B_k may be indefinite at any iteration, which means that the trust region bound may continue to be active for arbitrarily large k. Interestingly, however, it can be shown that the SR1 Hessian approximations tend to be positive definite most of the time. The precise result is that

$$\lim_{k\to\infty} \frac{\#\{\, j \,\big|\, 1 \leqslant j \leqslant k, B_j \text{ is positive semi-definite}\}}{k} = 1\,,$$

under the assumptions of the theorem above. This result holds regardless of whether the initial Hessian approximation is positive definite or not.

§6.4 Convergence Analysis

Note that the BFGS method does not require the boundedness assumption ③ to hold. As we have mentioned already, the SR1 update does not necessarily maintain positive definiteness of the Hessian approximations B_k. In practice, B_k may be indefinite at any iteration, which means that the trust region bound may continue to be active for arbitrarily large k. Interestingly, however, it can be shown that the SR1 Hessian approximations tend to be positive definite most of the time. The precise result is that

$$\lim_{k\to\infty} \frac{\#\{\, j \,\big|\, 1 \leqslant j \leqslant k,\, B_j \text{ is positive semi-definite} \}}{k} = 1\,,$$

under the assumptions of the theorem above. This result holds regardless of whether the initial Hessian approximation is positive definite or not.