Problem 5.1: Newton's Method for a Quadratic Problem

Consider the following quadratic optimization problem

$$\min_{x}.\ f(x) = x^\top Ax \tag{1}$$

with variable vector $x \in \mathbb{R}^n$, and where $A \in \mathbb{R}^{n\times n}$ is a positive semidefinite (PSD) matrix.

(a) Not all PSD matrices are symmetric. Show that if the PSD matrix A in (1) is not symmetric, we can always replace it with a symmetric PSD matrix B such that $x^\top Ax = x^\top Bx$.

(b) Show that Newton's method converges in one iteration when applied to the problem (1).

(c) Show that Newton's method converges in one iteration when applied to the following quadratic problem with an additional linear term:

$$\min_{x}.\ f(x) = x^\top Ax - b^\top x \tag{2}$$

with variables $x \in \mathbb{R}^n$. Assume that $A \in \mathbb{R}^{n\times n}$ is a symmetric PSD matrix and $b \in \mathbb{R}^n$.

Solution.

(a) If $A \in \mathbb{R}^{n\times n}$ is PSD but not symmetric, we can replace it with a symmetric PSD matrix $B \in \mathbb{R}^{n\times n}$ such that the following holds:

$$x^\top Ax = x^\top Bx$$

To this end, let us write A as a sum of the following two terms

$$A = \frac{A + A^\top}{2} + \frac{A - A^\top}{2}$$

Thus, the quadratic form $x^\top Ax$ becomes

$$\begin{aligned} x^\top Ax &= x^\top \left(\frac{A + A^\top}{2} + \frac{A - A^\top}{2}\right) x \\ &= \underbrace{x^\top \left(\frac{A + A^\top}{2}\right) x}_{\text{term 1}} + \underbrace{x^\top \left(\frac{A - A^\top}{2}\right) x}_{\text{term 2}} \end{aligned} \tag{3}$$

Let us evaluate term 2 in expression (3). We get

$$\begin{aligned} \frac{1}{2}x^\top(A - A^\top)x &= \frac{1}{2}(x^\top A - x^\top A^\top)x \\ &= \frac{1}{2}(x^\top Ax - x^\top A^\top x) \\ &= \frac{1}{2}(x^\top Ax - x^\top Ax) = 0 \end{aligned}$$

As term 2 evaluates to zero, it can be dropped completely from (3), which then becomes:

$$x^\top Ax = x^\top \left(\frac{A + A^\top}{2}\right) x \tag{4}$$

Now, we can notice that

$$B = \frac{A + A^\top}{2}$$

in (4) is a symmetric matrix. Using this notation, we can rewrite (4) as

$$x^\top Ax = x^\top Bx \tag{5}$$

Therefore, B is indeed positive semidefinite based on the equivalence (5) (this holds because A is PSD by definition). To verify this from the expression (4), we can write

$$\begin{aligned} x^\top Bx &= \frac{1}{2}x^\top(A + A^\top)x \\ &= \frac{1}{2}(x^\top A + x^\top A^\top)x \\ &= \frac{1}{2}(x^\top Ax + x^\top A^\top x) \\ &= \frac{1}{2}(x^\top Ax + x^\top Ax) \\ &= \frac{1}{2}(2x^\top Ax) \\ &= x^\top Ax \\ &\geq 0 \end{aligned}$$

which holds for all $x \in \mathbb{R}^n$ since A is PSD. Thus, we can always transform any quadratic form $x^\top Ax$ with a PSD matrix A to one where $A \in \mathbb{R}^{n\times n}$ is PSD and symmetric.

(b) Let $f(x) = x^\top Ax$. Recall that the update rule for Newton's method is defined as

$$x_{k+1} = x_k - H^{-1}(x_k)\nabla f(x_k) \tag{6}$$

By computing the gradient $\nabla f(x)$ and Hessian $H(x)$ of $f(x)$, we get (see, e.g. equation (93) in the Matrix Cookbook)

$$\begin{aligned} \nabla f(x) &= \nabla(x^\top Ax) \\ &= Ax + A^\top x \\ &= Ax + Ax \\ &= 2Ax \end{aligned} \tag{7}$$

and

$$H(x) = \nabla^2 f(x) = A + A^\top = 2A \tag{8}$$

where we used the fact that A is symmetric. Now, if we substitute (7) and (8) computed at some iteration k to (6), we get

$$x_{k+1} = x_k - A^{-1}Ax_k = x_k - x_k = 0 \tag{9}$$

Now, if we start at any point $x_0 \in \mathbb{R}^n$, the update rule (9) gives us $x_1 = 0$ in one iteration. By substituting $x_1 = 0$ to the gradient (7), we get

$$\nabla f(x_1) = 2Ax_1 = 0$$

Since $f(x) = x^\top Ax$ is convex (as $A \succeq 0$), the first order condition $\nabla f(x_1) = 0$ is both necessary and sufficient for global optimality. Thus, we obtain the optimal solution $x_1 = 0$ in one iteration regardless of the initial starting point $x_0 \in \mathbb{R}^n$.

(c) Let $f(x) = x^\top Ax - b^\top x$. Computing the gradient and Hessian of $f(x)$ in (2), we get

$$\begin{aligned} \nabla f(x) &= \nabla(x^\top Ax - b^\top x) \\ &= Ax + A^\top x - b \\ &= Ax + Ax - b \\ &= 2Ax - b \end{aligned} \tag{10}$$

and

$$H(x) = \nabla^2 f(x) = A + A^\top = 2A \tag{11}$$

where we used the fact that A is symmetric. Now, assume that we start with a random initial point $x_0 \in \mathbb{R}^n$. By substituting the gradient (10) and Hessian (11) computed ad x_0 to (6), we get

$$\begin{aligned}
x_1 &= x_0 - \frac{1}{2}A^{-1}(2Ax_0 - b) \\
&= x_0 - A^{-1}Ax_0 + \frac{1}{2}A^{-1}b = 0 \\
&= x_0 - x_0 + \frac{1}{2}A^{-1}b \\
&= \frac{1}{2}A^{-1}b
\end{aligned} \tag{12}$$

Now, by substituting $x_1 = (1/2)A^{-1}b$ to the gradient (10), we get

$$\begin{aligned}
\nabla f(x_1) &= 2A(1/2)A^{-1}b - b \\
&= AA^{-1}b - b \\
&= b - b \\
&= 0
\end{aligned}$$

Since $f(x) = x^\top Ax - b^\top x$ is convex (as $A \succeq 0$), the first order condition $\nabla f(x_1) = 0$ is both necessary and sufficient for global optimality, and we get the optimal solution $x_1 = (1/2)A^{-1}b$ in one iteration regardless of the starting point $x_0 \in \mathbb{R}^n$.

Problem 5.2: Affine Invariance of Newton's Method

Consider the following unconstrained optimization problem

$$\min_x . \; f(x) \tag{13}$$

where $f : \mathbb{R}^n \to \mathbb{R}$ is a twice differentiable function. Show that Newton's method applied to problem (13) is *affine invariant*, meaning that the progress of Newton's method is independent of affine transformations of the original problem (e.g., scaling, translation, and rotation).

Solution.

Recall that the update rule in Newton's method is

$$x_{k+1} = x_k - \nabla^2 f(x_k)^{-1}\nabla f(x_k) \tag{14}$$

Note that we use $\nabla^2 f(x)$ to represent the Hessian of f (as this allows to differentiate from the Hessian of g, $\nabla^2 g(x)$). Let $x = Ay + b$ where $A \in \mathbb{R}^{n\times n}$ is nonsingular and $b \in \mathbb{R}^n$. Define the function $g : \mathbb{R}^n \to \mathbb{R}$ as

$$g(y) = f(Ay + b) = f(x)$$

By computing the gradient and Hessian of $g(y)$, we get

$$\nabla g(y) = A^\top \nabla f(Ay + b) \quad \text{and} \quad \nabla^2 g(y) = A^\top \nabla^2 f(Ay + b)A$$

The Newton update of g at y_k is

$$\begin{aligned}
y_{k+1} &= y_k - \nabla^2 g(y_k)^{-1}\nabla g(y_k) \\
&= y_k - (A^\top \nabla^2 f(Ay_k + b)A)^{-1}A^\top \nabla f(Ay_k + b) \\
&= y_k - A^{-1}\nabla^2 f(Ay_k + b)^{-1}(A^\top)^{-1}A^\top \nabla f(Ay_k + b) \\
&= y_k - A^{-1}\nabla^2 f(Ay_k + b)^{-1}\nabla f(Ay_k + b)
\end{aligned} \tag{15}$$

By first multiplying both sides of (15) with the scaling matrix A, and then adding the vector b to both sides we get

$$Ay_{k+1} + b = Ay_k + b - \nabla^2 f(Ay_k + b)^{-1} \nabla f(Ay_k + b) \tag{16}$$

which is exactly the same as

$$x_{k+1} = x_k - \nabla^2 f(x_k)^{-1} \nabla f(x_k) \tag{17}$$

since $x = Ay + b$. Therefore, the progress of Newton's method is independent of problem scaling, as both (16) and (17) have exactly the same progress. To make this point extra clear, if we apply Newton's method to g at y_k to obtain y_{k+1}, we can compute the corresponding x_{k+1} simply by applying the affine transformation

$$x_{k+1} = Ay_{k+1} + b$$

and we would get exactly the same x_{k+1} if we applied Newton's method to f at x_k.