Exercises in Optimization
(ACM 40990 / ACM41030)

Dr Lennon Ó Náraigh

Exercises #2

Exercises #2 - More on Line-search Methods

1. In the notes (Chapter 6), it is shown that the Newton method satisfies:

$$\|\boldsymbol{x}_{k+1}-\boldsymbol{x}_*\|_2 \le C\|\boldsymbol{x}_k-\boldsymbol{x}_*\|_2^2 \text{ whenever } \|\boldsymbol{x}_k-\boldsymbol{x}_*\|_2 < \delta.$$

If we choose

$$\|\boldsymbol{x}_0-\boldsymbol{x}_*\|_2 < \delta, \text{ and } \|\boldsymbol{x}_0-\boldsymbol{x}_*\| < \frac{1}{2C},$$

then

$$\frac{\|\boldsymbol{x}_1-\boldsymbol{x}_*\|_2}{\|\boldsymbol{x}_0-\boldsymbol{x}_*\|_2} \le C\|\boldsymbol{x}_0-\boldsymbol{x}_*\|_2 \le \tfrac{1}{2}.$$

The aim of this exercise is to show that these inequalities give rise to the following important result:

$$\frac{\|\boldsymbol{x}_k-\boldsymbol{x}_*\|_2}{\|\boldsymbol{x}_0-\boldsymbol{x}_*\|_2} \le \frac{1}{2^{2^k-1}}. \tag{1}$$

The proof of the inequality (1) can be obtained by the following sequence of steps:

(a) Write down inequalities for

$$\|\boldsymbol{x}_1-\boldsymbol{x}_*\|_2 \le \frac{1}{2^{\cdots}}\|\boldsymbol{x}_0-\boldsymbol{x}_*\|_2, \qquad \|\boldsymbol{x}_2-\boldsymbol{x}_*\|_2 \le \frac{1}{2^{\cdots}}\|\boldsymbol{x}_0-\boldsymbol{x}_*\|_2,$$
$$\|\boldsymbol{x}_3-\boldsymbol{x}_*\|_2 \le \frac{1}{2^{\cdots}}\|\boldsymbol{x}_0-\boldsymbol{x}_*\|_2.$$

(b) Hence, guess that the general term satisfies

$$\|\boldsymbol{x}_k-\boldsymbol{x}_*\| \le \frac{1}{2^{p_k}}\|\boldsymbol{x}_0-\boldsymbol{x}_*\|_2,$$

where

$$p_k = 2p_{k-1}+1. \tag{2}$$

(c) Equation (2) is a first-order **difference equation** with general solution $p_k = A + B\lambda^n$, where A, B, and λ are constants to be determined. Hence, show that p_k satisfies:

$$p_k = 2^k - 1, \qquad k > 1$$

with $p_1 = 1$.

(d) Conclude that

$$\|\boldsymbol{x}_k - \boldsymbol{x}_*\|_2 \leq \frac{1}{2^{2^k-1}}\|\boldsymbol{x}_0 - \boldsymbol{x}_*\|_2,$$

and hence,

$$\lim_{k\to\infty} \|\boldsymbol{x}_k - \boldsymbol{x}_*\|_2 = 0.$$

For part (a) we have:

$$\begin{aligned}
\|\boldsymbol{x}_1 - \boldsymbol{x}_*\|_2 &\leq C\|\boldsymbol{x}_0 - \boldsymbol{x}_*\|_2^2, \\
&\leq (C\|\boldsymbol{x}_0 - \boldsymbol{x}_*\|_2)\,\|\boldsymbol{x}_0 - \boldsymbol{x}_*\|_2, \\
&\leq \tfrac{1}{2}\|\boldsymbol{x}_0 - \boldsymbol{x}_*\|_2.
\end{aligned}$$

Also,

$$\begin{aligned}
\|\boldsymbol{x}_2 - \boldsymbol{x}_*\|_2 &\leq C\|\boldsymbol{x}_1 - \boldsymbol{x}_*\|_2^2, \\
&\leq C\left(\tfrac{1}{2}\|\boldsymbol{x}_0 - \boldsymbol{x}_*\|_2\right)\left(\tfrac{1}{2}\|\boldsymbol{x}_0 - \boldsymbol{x}_*\|_2\right), \\
&\leq \tfrac{1}{2^2}\left(C\|\boldsymbol{x}_0 - \boldsymbol{x}_*\|_2\right)\|\boldsymbol{x}_0 - \boldsymbol{x}_*\|_2, \\
&\leq \tfrac{1}{2^3}\|\boldsymbol{x}_0 - \boldsymbol{x}_*\|_2.
\end{aligned}$$

Finally,

$$\begin{aligned}
\|\boldsymbol{x}_3 - \boldsymbol{x}_*\|_2 &\leq C\|\boldsymbol{x}_2 - \boldsymbol{x}_*\|_2^2, \\
&\leq C\left(\tfrac{1}{2^3}\|\boldsymbol{x}_0 - \boldsymbol{x}_*\|_2\right)\left(\tfrac{1}{2^3}\|\boldsymbol{x}_0 - \boldsymbol{x}_*\|_2\right), \\
&\leq \tfrac{1}{2^6}\left(C\|\boldsymbol{x}_0 - \boldsymbol{x}_*\|_2\right)\|\boldsymbol{x}_0 - \boldsymbol{x}_*\|_2, \\
&\leq \tfrac{1}{2^7}\|\boldsymbol{x}_0 - \boldsymbol{x}_*\|_2.
\end{aligned}$$

For part (b), we guess the pattern, p_k is the power of two in the general term, and we have:

$$\begin{aligned}
p_1 &= 1, \\
p_2 &= 3, \\
p_3 &= 7,
\end{aligned}$$

so we guess:

$$p_k = 2p_{k-1} + 1 \tag{3}$$

for $k > 1$, and $p_k = 1$ for $k = 1$.

For part (c), we substitute the trial solution $p_k = A + B\lambda^k$ into the difference equation (3). We have:

$$A + B\lambda^k = 2\left(A + B\lambda^{k-1}\right) + 1.$$

We equate coefficients to get $A = -1$ and $\lambda = 2$, hence

$$p_k = B(2^k) - 1.$$

We have $p_1 = 1$, hence $B = 1$ also, hence:

$$p_k = 2^k - 1,$$

hence (part (d)),

$$\|\boldsymbol{x}_k - \boldsymbol{x}_*\|_2 \leq \frac{1}{2^{2^k-1}}\|\boldsymbol{x}_0 - \boldsymbol{x}_*\|_2,$$

hence

$$\lim_{k\to\infty} \|\boldsymbol{x}_k - \boldsymbol{x}_*\|_2 = 0,$$

as required.

2. Show that if $0 < c_2 < c_1 < 1$, there may be no step lengths that satisfy the Strong Wolfe conditions.

   Hint: Consider the quadratic function

   $$\phi(\alpha) = a + b\alpha + c\alpha^2,$$

   where $b < 0$ and $c > 0$.

We compute $\phi(0) = a$ and $\phi'(0) = b$. SW1 requires $\phi(\alpha) \le a + c_1 b\alpha$, hence:

$$b\alpha + c\alpha^2 \le c_1 b\alpha,$$

hence $\alpha = 0$ or

$$\alpha = -b(1 - c_1)/c.$$

Identify $\alpha_1 = |b|(1 - c_1)/c$ with $0 < c_1 < 1$. Thus, for SWC1 to hold we require:

$$0 \le \alpha \le \alpha_1 = \frac{|b|(1 - c_1)}{c}.$$

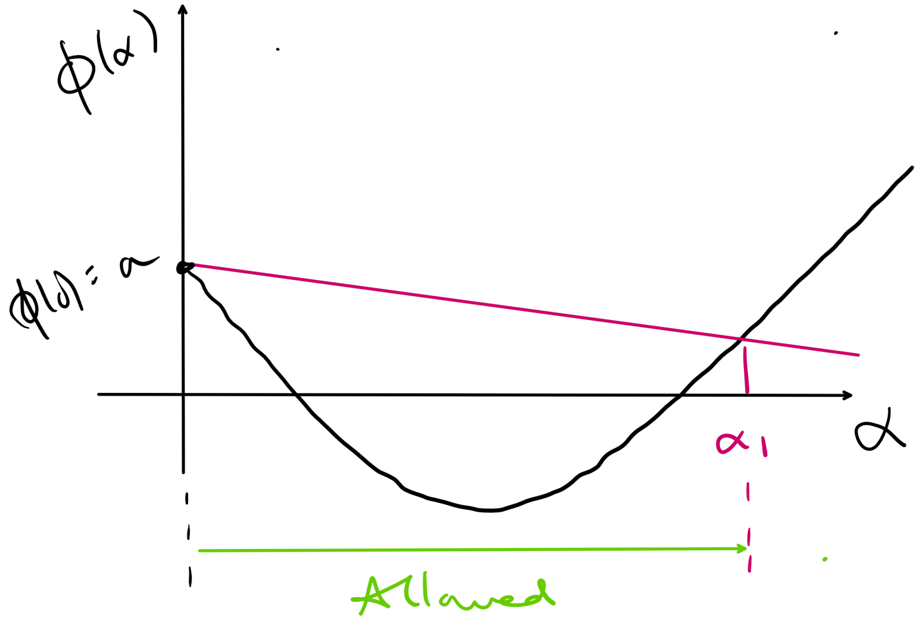


Figure 1: The idea behind SWC1 in the case of a quadratic function $\phi(\alpha) = a + b\alpha + c\alpha^2$

For SWC2 to hold, we require $|\phi'(\alpha)| \le c_2|b|$, hence $|b + 2c\alpha| \le c_2|b|$. This is a quadratic inequality in disguise:

$$b^2 + 4bc\alpha + 4c^2\alpha^2 \le c_2^2 b^2.$$

We have:

$$4c^2\alpha^2 + 4bc\alpha + b^2(1 - c_2^2) = 0.$$

Hence:

$$
\begin{aligned}
\alpha &= \frac{-4bc \pm \sqrt{16b^2c^2 - 16c^2b^2(1-c_2^2)}}{8c^2},\\
&= \frac{-4bc \pm \sqrt{16c^2b^2c_2^2}}{8c^2},\\
&= \frac{-4bc \pm 4|b|cc_2}{8c^2},\\
&= \frac{4|b|c \pm 4|b|cc_2}{8c^2},\\
&= \frac{|b|(1\pm c_2)}{2c},\\
&= \alpha_\pm.
\end{aligned}
$$

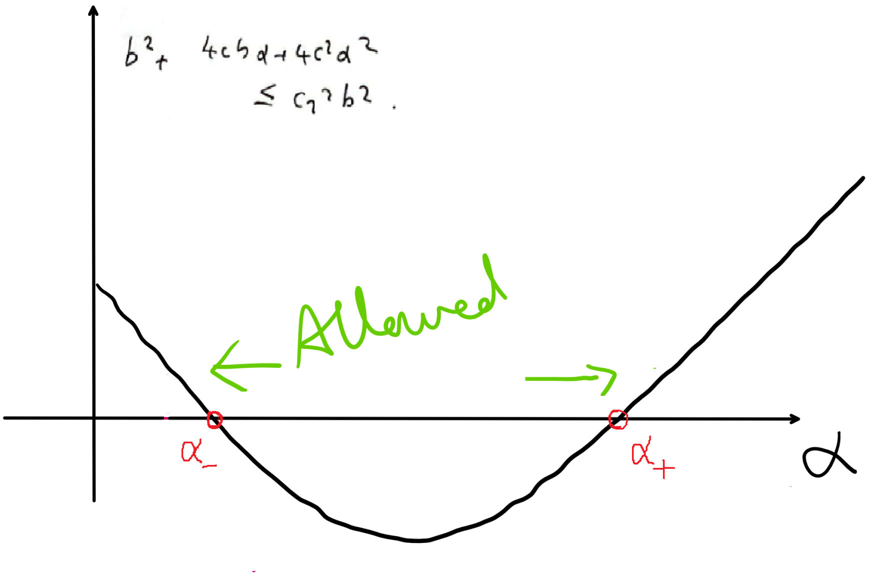


Figure 2: The idea behind SWC2 in the case of a quadratic function $\phi(\alpha) = a + b\alpha + c\alpha^2$

Referring to Figures 1 and 2, there is a contradiction (and hence, the SWCs don't hold) if

$$\alpha_- > \alpha_1, \tag{4}$$

hence, if:

$$\frac{|b|(1-c_2)}{2c} > \frac{|b|(1-c_1)}{c}. \tag{5}$$

Hence, the SWCs do not hold if

$$c_2 < 2c_1.$$

However, we are given that $0 < c_2 < c_1 < 1$, then it is certainly the case that

$$c_2 < c_1 < 2c_1. \tag{6}$$

Hence, reasoning backwards, if $0 < c_2 < c_1 < 1$, then inequality (6) holds, then inequality (5) holds also, and then so does inequality (4), and hence, the SWCs cannot be satisfied.

3. Consider the one-dimensional function

$$\phi(\alpha) = f(\boldsymbol{x}_k + \alpha \boldsymbol{p}_k),$$

where $\boldsymbol{p}_k$ is a descent direction – that is, $\phi'(0) < 0$ – so that our search can be confined to positive values of α. Find the value that minimizes $\phi(\alpha)$ in the case where the cost function is quadratic, specifically:

$$f(\boldsymbol{x}) = \langle \boldsymbol{a}, \boldsymbol{x} \rangle + \tfrac{1}{2}\langle \boldsymbol{x}, B\boldsymbol{x} \rangle, \tag{7}$$

where $\boldsymbol{a} \in \mathbb{R}^n$ and $B \in \mathbb{R}^{n \times n}$.

We have:

$$\begin{aligned}
\phi(\alpha) &= f(\boldsymbol{x}_k + \alpha \boldsymbol{p}_k), \\
&= \langle \boldsymbol{a}, \boldsymbol{x}_k + \alpha \boldsymbol{p}_k \rangle + \tfrac{1}{2}\langle \boldsymbol{x}_k + \alpha \boldsymbol{p}_k, B\boldsymbol{x}_k + \alpha B \boldsymbol{p}_k \rangle, \\
&= \underbrace{\langle \boldsymbol{a}, \boldsymbol{x}_k \rangle} + \alpha \langle \boldsymbol{a}, \boldsymbol{p}_k \rangle + \underbrace{\tfrac{1}{2}\langle \boldsymbol{x}_k, B\boldsymbol{x}_k \rangle} + \tfrac{1}{2}\alpha \langle \boldsymbol{p}_k, B\boldsymbol{x}_k \rangle + \tfrac{1}{2}\alpha \langle \boldsymbol{x}_k, B\boldsymbol{p}_k \rangle + \tfrac{1}{2}\alpha^2 \langle \boldsymbol{p}_k, B\boldsymbol{p}_k \rangle.
\end{aligned}$$

Continuing thus, we have:

$$\begin{aligned}
\phi(\alpha) &= \underbrace{f(\boldsymbol{x}_k)} + \alpha \langle \boldsymbol{a}, \boldsymbol{p}_k \rangle + \tfrac{1}{2}\alpha \langle \boldsymbol{p}_k, B\boldsymbol{x}_k \rangle + \tfrac{1}{2}\alpha \langle \boldsymbol{x}_k, B\boldsymbol{p}_k \rangle + \tfrac{1}{2}\alpha^2 \langle \boldsymbol{p}_k, B\boldsymbol{p}_k \rangle, \\
&= f(\boldsymbol{x}_k) + \alpha \langle \boldsymbol{a}, \boldsymbol{p}_k \rangle + \tfrac{1}{2}\alpha \langle \boldsymbol{p}_k, B\boldsymbol{x}_k \rangle + \tfrac{1}{2}\alpha \langle B^T \boldsymbol{x}_k, \boldsymbol{p}_k \rangle + \tfrac{1}{2}\alpha^2 \langle \boldsymbol{p}_k, B\boldsymbol{p}_k \rangle, \\
&= f(\boldsymbol{x}_k) + \alpha \langle \boldsymbol{a}, \boldsymbol{p}_k \rangle + \tfrac{1}{2}\alpha \langle \boldsymbol{p}_k, B\boldsymbol{x}_k \rangle + \tfrac{1}{2}\alpha \langle B \boldsymbol{x}_k, \boldsymbol{p}_k \rangle + \tfrac{1}{2}\alpha^2 \langle \boldsymbol{p}_k, B\boldsymbol{p}_k \rangle, \\
&= f(\boldsymbol{x}_k) + \alpha \langle \boldsymbol{a}, \boldsymbol{p}_k \rangle + \tfrac{1}{2}\alpha \langle \boldsymbol{p}_k, B\boldsymbol{x}_k \rangle + \tfrac{1}{2}\alpha \langle B^T \boldsymbol{x}_k, \boldsymbol{p}_k \rangle + \tfrac{1}{2}\alpha^2 \langle \boldsymbol{p}_k, B\boldsymbol{p}_k \rangle, \\
&= f(\boldsymbol{x}_k) + \alpha \langle \boldsymbol{a}, \boldsymbol{p}_k \rangle + \alpha \langle \boldsymbol{p}_k, B\boldsymbol{x}_k \rangle + \tfrac{1}{2}\alpha^2 \langle \boldsymbol{p}_k, B\boldsymbol{p}_k \rangle, \\
&= f(\boldsymbol{x}_k) + \alpha \langle \boldsymbol{p}_k, a + B_k \boldsymbol{x}_k + \tfrac{1}{2}\alpha^2 \langle \boldsymbol{p}_k, B\boldsymbol{p}_k \rangle.
\end{aligned}$$

Hence,

$$\phi'(\alpha) = \langle \boldsymbol{p}_k, B\boldsymbol{x}_k + \boldsymbol{a} \rangle + \alpha \langle \boldsymbol{p}_k, B\boldsymbol{p}_k \rangle.$$

When $\phi'(\alpha) = 0$ we have:

$$\alpha = -\frac{\langle \boldsymbol{p}_k, B\boldsymbol{x}_k + \boldsymbol{a} \rangle}{\langle \boldsymbol{p}_k, B\boldsymbol{p}_k \rangle},$$

or

$$\alpha = -\frac{\langle \boldsymbol{p}_k, \nabla f_k \rangle}{\langle \boldsymbol{p}_k, B\boldsymbol{p}_k \rangle},$$

4. Consider the steepest decent method with exact line searches applied to the convex quadratic function in Equation (7).
   (a) Show that if the initial point is such that $\boldsymbol{x}_0 - \boldsymbol{x}_*$ is parallel to an eigenvector of B, then the steepest descent method will find the solution in one step.
   (b) Show that the Newton method always converges in exactly one step when the cost function is quadratic, i.e. takes the form (7).

For Part (a) we have:

$$\begin{aligned}\nabla f_0 &= B\boldsymbol{x}_0 + \boldsymbol{a},\\ &= B\boldsymbol{x}_0 + (-B\boldsymbol{x}_*),\\ &= B(\boldsymbol{x}_0 - \boldsymbol{x}_*).\end{aligned}$$

But $\boldsymbol{x}_0 - \boldsymbol{x}_*$ is parallel to an eigenvector (eigenvalue: λ), hence:

$$\begin{aligned}\nabla f_0 &= B(\boldsymbol{x}_0 - \boldsymbol{x}_*),\\ &= \lambda(\boldsymbol{x}_0 - \boldsymbol{x}_*).\end{aligned}$$

From the previous question, we have the value for the exact stepsize, and we are using the SD algorithm, hence $\boldsymbol{p}_0 = -\nabla f_0$. Hence,

$$\begin{aligned}\alpha &= -\frac{\langle \boldsymbol{p}_0, \nabla f_0\rangle}{\langle \boldsymbol{p}_0, B\boldsymbol{p}_0\rangle},\\ &= \frac{\langle \nabla f_0, \nabla f_0\rangle}{\langle \nabla f_0, B\nabla f_0\rangle},\\ &= \frac{\lambda^2\langle \boldsymbol{x}_0 - \boldsymbol{x}_*, \boldsymbol{x}_0 - \boldsymbol{x}_*\rangle}{\lambda^2\langle \boldsymbol{x}_0 - \boldsymbol{x}_*, B(\boldsymbol{x}_0 - \boldsymbol{x}_*)\rangle},\\ &= \frac{1}{\lambda}.\end{aligned}$$

Finally, we have the steepest-descent step:

$$\begin{aligned}\boldsymbol{x}_1 &= \boldsymbol{x}_0 - \alpha\nabla f_0,\\ &= \boldsymbol{x}_0 - \frac{1}{\lambda}\left[\lambda(\boldsymbol{x}_0 - \boldsymbol{x}_*)\right],\\ &= \boldsymbol{x}_*,\end{aligned}$$

and thus, the SD algorithm converges in one step.

For Part (b) we have:

$$\begin{aligned}\boldsymbol{x}_1 &= \boldsymbol{x}_0 - B^{-1}\nabla f_0,\\ &= \boldsymbol{x}_0 - B^{-1}\left[B\boldsymbol{x}_0 + \boldsymbol{a}\right],\\ &= -B^{-1}\boldsymbol{a},\\ &= \boldsymbol{x}_*,\end{aligned}$$

and thus, the Newton algorithm converges in one step.

5. Consider the optimization problem,

$$\min f(\boldsymbol{x}), \qquad f(\boldsymbol{x}) = \langle \boldsymbol{a}, \boldsymbol{x} \rangle + \tfrac{1}{2}\langle \boldsymbol{x}, B\boldsymbol{x} \rangle,$$

where now B is a specific 10×10 matrix and $\boldsymbol{a}$ is a specific 10×1 column vector. The numerical values of these arrays can be found in the spreadsheet `OP_10x10.csv`:

- The spreadsheet contains a 10×1 array which corresponds to the vector $\boldsymbol{a}$;
- The spreadsheet contains a 10×10 array B_0.

The array B is obtained from B_0 by the following sequence of steps:

(i) Symmetrize B_0:

$$B_0 \to (B_0 + B_0^T)/2;$$

(ii) Scale B_0:

$$B_0 \to B_0 / \max(|B_0|)$$

(iii) Generate a positive-definite matrix:

$$B_0 \to (B_0^T)B_0.$$

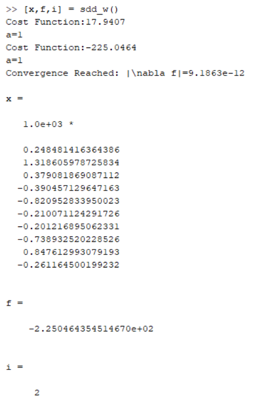
The end result of this sequence of operations is the matrix B.

Hence,

(a) Find the minimizer $\boldsymbol{x}_*$ numerically, using the steepest-descent and Newton algorithms.

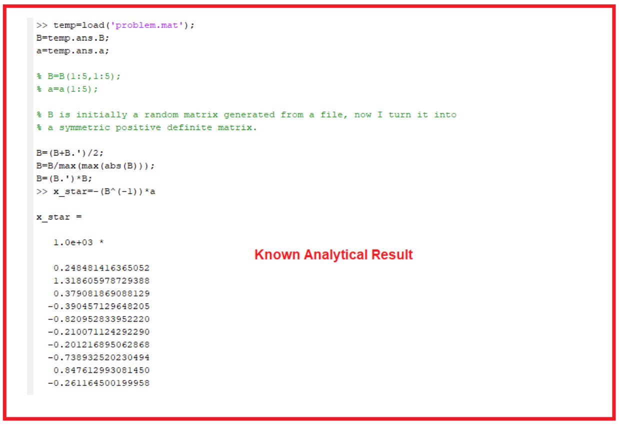
(b) Why is the convergence so poor in the case of the steepest-descent algorithm?

For part (a), we present sample codes in Matlab, these can be found in an accompanying folder (`OP_10x10`). The convergence is extremely slow in case of the SD method (for the given numerical parameters); results are not presented here. Instead, we go straight over to the Newton method, where we obtain the following results:



```
>> [x,f,i] = sdd_w()
Cost Function:17.9407
a=1
Cost Function:-225.0464
a=1
Convergence Reached: |\nabla f|=9.1863e-12

x =

   1.0e+03 *

   0.248481416364386
   1.318605978725834
   0.379081869087112
  -0.390457129647163
  -0.820952833950023
  -0.210071124291726
  -0.201216895062331
  -0.738932520228526
   0.847612993079193
  -0.261164500199232


f =

    -2.250464354514670e+02


i =

     2
```

Results using
Newton Method

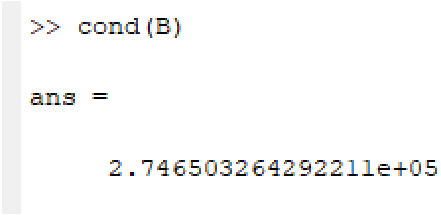


```
>> temp=load('problem.mat');
B=temp.ans.B;
a=temp.ans.a;

% B=B(1:5,1:5);
% a=a(1:5);

% B is initially a random matrix generated from a file, now I turn it into
% a symmetric positive definite matrix.

B=(B+B.')/2;
B=B/max(max(abs(B)));
B=(B.')*B;
>> x_star=-(B^(-1))*a

x_star =

   1.0e+03 *

   0.248481416365052
   1.318605978729388
   0.379081869088129
  -0.390457129648205
  -0.820952833952220
  -0.210071124292290
  -0.201216895062868
  -0.738932520230494
   0.847612993081450
  -0.261164500199958
```

Known Analytical Result

This can be checked, as we know the analytical value of the minimizer in this case, $\boldsymbol{x}_* = -B^{-1}\boldsymbol{a}$, this is also shown in the figure. The numerical and analytical solutions agree, confirming the correct implementation of the Newton method in this case.

For part (b), we look at the condition number of the matrix B:



```
>> cond(B)

ans =

     2.746503264292211e+05
```

This is greater than 10^5 meaning the maximum and minimum eigenvalues are orders of magnitude apart. From class notes, and for the SD method, we have:

$$\|\boldsymbol{x}_{k+1} - \boldsymbol{x}_*\|_B^2 \le \left(1 - \frac{1}{\kappa(B)}\right) \|\boldsymbol{x}_k - \boldsymbol{x}_*\|_B^2,$$

and with $\kappa(B) > 10^5$, we have:

$$\|\boldsymbol{x}_{k+1} - \boldsymbol{x}_*\|_B^2 \lesssim \|\boldsymbol{x}_k - \boldsymbol{x}_*\|_B^2,$$

leading to poor convergence of the SD method in this particular example.