HW5

Mingju Liu ml1499

Problem 1:

For this problem, I use the least square function as the unconstrained objective function.

$$\text{minimize } ||Ax - b||_2^2$$

From the Textbook, I know that the gradient descent algorithm,

Algorithm 9.3 *Gradient descent method.*

given a starting point $x \in \textbf{dom} f$.

repeat

1. $\Delta x := -\nabla f(x)$.
2. *Line search.* Choose step size t via exact or backtracking line search.
3. *Update.* $x := x + t\Delta x$.

until stopping criterion is satisfied.

And also the backtracking line search algorithm,

Algorithm 9.2 *Backtracking line search.*

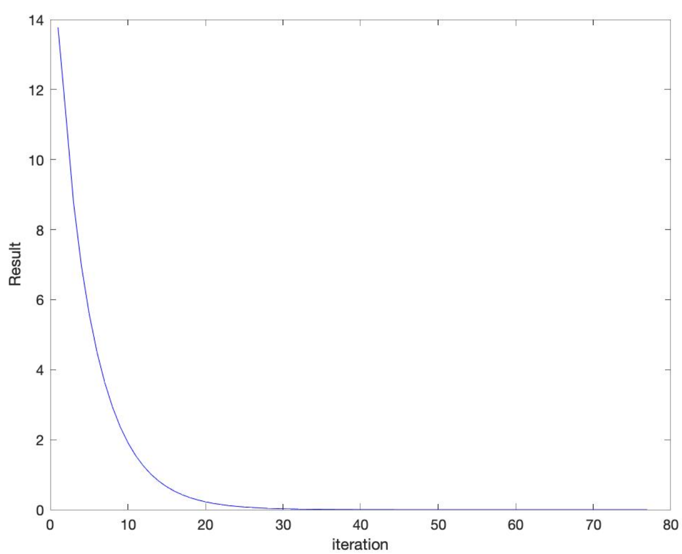
given a descent direction Δx for f at $x \in \textbf{dom} f$, $\alpha \in (0, 0.5)$, $\beta \in (0, 1)$.

$t := 1$.

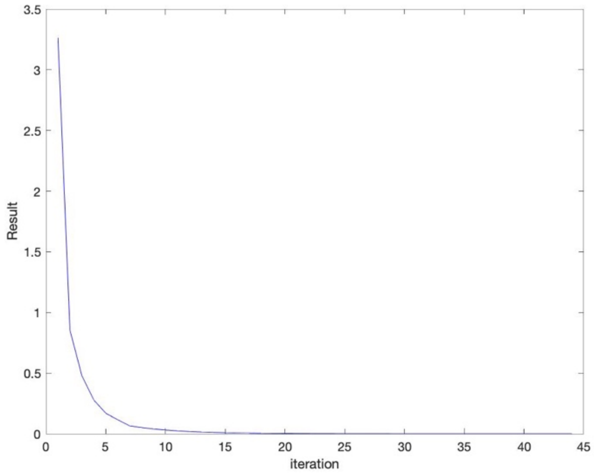
while $f(x + t\Delta x) > f(x) + \alpha t \nabla f(x)^T \Delta x$, $\quad t := \beta t$.

With implementation of these two algorithms in Matlab, I can obtain the results, with different positive definite matrix with different condition numbers. The results are as follows:

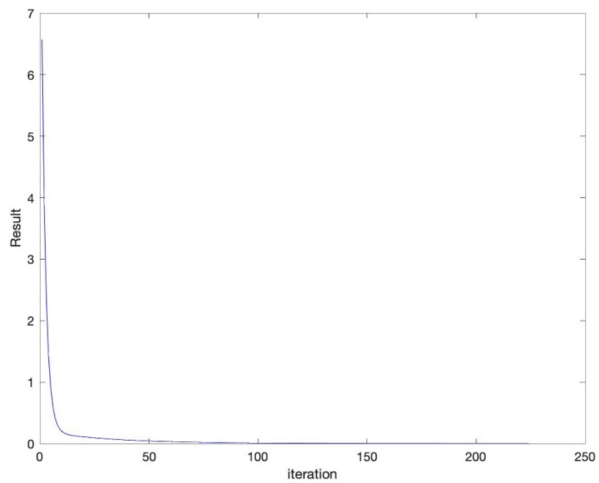
Condition number = 2:



Condition number = 3:



Condition number = 4:



Problem 2:

I added an equality constraint to the first objective function:

Px = Q,

Then I can implement the newton method from the textbook:

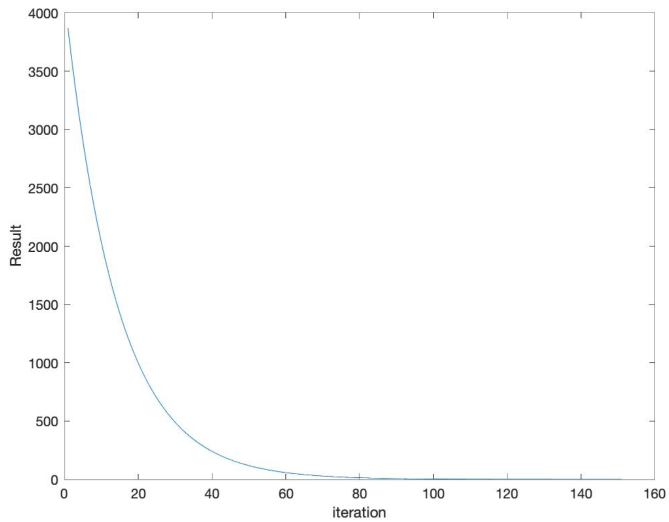
Algorithm 9.5 *Newton's method.*

given a starting point $x \in \mathbf{dom}\, f$, tolerance $\epsilon > 0$.

repeat

1. *Compute the Newton step and decrement.*
   $\Delta x_{\mathrm{nt}} := -\nabla^2 f(x)^{-1} \nabla f(x)$; $\quad \lambda^2 := \nabla f(x)^T \nabla^2 f(x)^{-1} \nabla f(x)$.
2. *Stopping criterion.* **quit** if $\lambda^2/2 \le \epsilon$.
3. *Line search.* Choose step size t by backtracking line search.
4. *Update.* $x := x + t\Delta x_{\mathrm{nt}}$.

And the result will be,



4000
3500
3000
2500
2000
1500
1000
500
0
Result
0
20
40
60
80
100
120
140
160
iteration