Advanced Topics in Numerical Analysis: Numerical Optimization

CSCI–GA.2945-003
MATH–GA.2012-003
Instructor: Margaret Wright
Computer Science Department
New York University, Spring 2019

Homework Assignment 5
Assigned Tuesday, April 23, 2019 (not a lecture day)
Due 11:59pm, Monday, May 6, 2019 (not a lecture day)

Exercise 5.1. Consider the following local quadratic model of a twice-continuously differentiable objective function \( f(x) \) at the point \( x_k \):

\[
q_k(x) = f_k + g_k^T(x - x_k) + \frac{1}{2}(x - x_k)^T B_k(x - x_k),
\]

where \( f_k = f(x_k) \), \( g_k = \nabla f(x_k) \), and \( B_k \) is symmetric. When we move from \( x_k \) to the new point \( x_k + d \), the change in the quadratic model is

\[
\Delta_k(d) = q_k(x_k + d) - q_k(x_k) = g_k^T d + \frac{1}{2}d^T B_k d.
\]

Assuming that \( d = \alpha p_k \), where the search direction \( p_k \) satisfies \( B_k p_k = -g_k \), show that \( \Delta_k(\alpha p_k) < 0 \) if \( B_k \) is positive definite and \( 0 < \alpha < 2 \).

Exercise 5.2. This problem involves writing code to perform a BFGS method for unconstrained minimization of a smooth function \( f(x) \). Each new iterate is given by \( x_{k+1} = x_k + \alpha_k p_k \), where \( \alpha_k \geq 0 \). The code should:

- Perform the BFGS updates on approximations \( \{B_k\} \) to the Hessian (not approximations to the inverse Hessian).
- Compute the search direction by solving \( B_k p_k = -g_k \) for \( p_k \) from scratch at each iteration, using backslash, where \( g_k = \nabla f(x_k) \). When \( y^T_k s_k > 0 \), generate \( B_{k+1} \) explicitly by adding the associated rank-two change to \( B_k \). If \( y^T_k s_k \leq 0 \), don’t update the matrix.
- Start with \( B_0 = I \) and stop when \( \|g_k\| \) is less than \( \text{ftol} \) or else after \( \text{maxit} \) iterations. (Try \( \text{ftol} = 1.0\times10^{-6} \) and \( \text{maxit} = 20 \).
- Print the optimal solution \( x^* \) and \( f(x^*) \) when your code stops.
- At each iteration, print \( k, x_k, f_k, \|g_k\|, \alpha_k, B_k, \|H - B_k\| \), and a message if the update was skipped.

There should be two options for the line search:

**LS1.** The value of \( \alpha_k \) at each iteration is obtained from a backtracking line search that starts with \( \alpha = 1 \) and uses reduction factor \( \gamma_c = \frac{1}{2} \), terminating when the Armijo condition is satisfied with \( \eta_s = 0.001 \). The line search terminates with a failure if \( \alpha \) needs to be reduced more than 10 times.

**LS2.** When minimizing a quadratic function \( q(x) \) whose (known) Hessian \( H \) is symmetric and positive definite, \( \alpha_k \) at iteration \( k \) is the exact step to the minimizer of \( q \) along \( p_k \), i.e.,

\[
\alpha_k = -\frac{g_k^T p_k}{p_k^T H p_k}, \quad \text{where } g_k \text{ is the gradient of } q \text{ at } x_k.
\]
Here are the two functions of interest:

\[
\hat{f}(x) = 5 \left( x_2 - x_1^2 \right)^2 + (x_1 - 1)^2; \\
\bar{f}(x) = x_1 + x_2 + x_3 + x_4 + \frac{1}{2} (2x_1^2 + x_2^2 + 10^{-1}x_3^2 + 10^{-3}x_4^2).
\]

(5.1) (5.2)

(a) Apply your code with line search option LS1 to minimize function (5.1) with \(x_0 = (0, -1)\).

(i) Give the exact solution \(x^*\) and the Hessian \(H(x^*)\).

(ii) Comment on interesting aspects of the iterates, especially \(\alpha_k\), and explain why they are interesting.

(iii) Do you observe superlinear convergence? Explain.

(iv) Does \(B_k\) seem to be converging to \(H(x^*)\)? Explain.

(b) Apply your code with line search option LS1 to minimize function (5.2) with \(x_0 = (-1, 0, 1, 1)\).

(i) Give the exact solution \(x^*\) and the Hessian \(H(x^*)\).

(ii) Comment on interesting aspects of the iterates, especially \(\alpha_k\), and explain why they are interesting.

(iii) Do you observe superlinear convergence? Explain.

(iv) Does \(B_k\) seem to be converging to \(H(x^*)\)? Explain.

(c) Apply your code with line search option LS2 to minimize function (5.2) with \(x_0 = (-1, 0, 1, 1)\).

(i) Comment on interesting aspects of the iterates, especially \(\alpha_k\), and explain why they are interesting.

(ii) Do you observe superlinear convergence? Explain.

(iii) Does \(B_k\) seem to be converging to \(H(x^*)\)? Explain.

(d) Apply your code with line search option LS2 to minimize function (5.2) with \(x_0 = (-\frac{1}{2}, 0, 1, 1)\).

Does \(B_k\) seem to be converging to the exact Hessian? If not, can you explain why?

Exercise 5.3. This problem involves trying (simplified versions of!) two approaches for solving a 2-variable nonlinear equality-constrained optimization problem of minimizing \(f(x)\) subject to \(c(x) = 0\), where \(c\) is a scalar-valued function.

Approach 1 involves using a pure Newton method to perform unconstrained minimization of the quadratic penalty function \(P_Q(x, \rho) = f(x) + \frac{1}{2}\rho \|c(x)\|^2\), so that the \(k\)th search direction \(p_k\) for a given value of \(\rho\) satisfies the Newton equations

\[
\nabla^2 P_Q(x_k, \rho) p_k = -\nabla P_Q(x_k, \rho).
\]

At \(x^*(\rho)\), an unconstrained minimizer of \(P_Q(x, \rho)\), the vector \(\lambda(\rho) = -\rho c(x^*(\rho))\) is an estimate of the Lagrange multiplier vector for the original problem (if Lagrange multipliers exist).

Approach 2, sometimes called the Newton-Lagrange method, is applicable when the problem satisfies a constraint qualification (so that Lagrange multipliers exist). In these circumstances, the optimal \(x\) and \(\lambda\) have the property that the \((n + m)\)-dimensional gradient of the Lagrangian function with respect to \(x\) and \(\lambda\), denoted by \(F(x, \lambda)\), is equal to zero. Hence the strategy is to use Newton’s method to find \((x, \lambda)\) satisfying the following system of nonlinear equations:

\[
F(x, \lambda) = \begin{pmatrix}
g(x) - J(x)^T \lambda \\
c(x)
\end{pmatrix} = 0.
\]

The associated Newton equations are

\[
\begin{pmatrix}
W(x_k, \lambda_k) & -J(x_k)^T \\
J(x_k) & 0
\end{pmatrix}
\begin{pmatrix}
p_k \\
\delta_k
\end{pmatrix} = -F(x_k, \lambda_k),
\]
where \( W(x_k, \lambda_k) \) is the Hessian of the Lagrangian function at \((x_k, \lambda_k)\), \( p_k \) is the step in \( x \), and \( \delta_k \) is the step in \( \lambda \). Note that there is no penalty parameter, but that the Lagrange multiplier is treated as an independent variable.

In this exercise, you are asked to apply Approach 1 for four values of the penalty parameter \((\rho = 1, 10, 100, 1000)\), starting at a specified point \( x_0 \) for the smallest \( \rho \). After optimizing the penalty function for the first penalty parameter, the Newton iterations to minimize the penalty function for the next value of \( \rho \) should begin at the last iterate for the previous \( \rho \), as in classical penalty function methods. Your program should terminate the Newton iterations for each value of \( \rho \) after maxit iterations or when \( \| \nabla P_0 \| \) is less than \( \text{ftol} \).

If you use the starting points given below, a pure Newton method should work well in both Approaches 1 and 2 (i.e., the step \( \alpha \) can safely be taken as 1). (You may wish to adapt code from earlier homework.)

The specific problem of interest is to minimize \( f(x) \) subject to \( c(x) = 0 \), with

\[
 f(x) = x_1^3 - x_1 x_2 \quad \text{and} \quad c(x) = \frac{5}{2} x_1^2 + \frac{1}{4} x_2^2 - \frac{7}{2} = 0.
\] (5.3)

The contours of \( f \) are shown in the figure, along with the boundary of the ellipse where the constraint is satisfied.

(a) Verify numerically that \( x^* = (-1, -2) \) satisfies the sufficient optimality conditions for problem (5.3). Give \( f(x^*) \). What is the optimal Lagrange multiplier \( \lambda^* \) and how did you obtain it?

(b) Consider the point \( \bar{x} = (1, -2) \). Show that \( \bar{x} \) is a first-order KKT point for this problem. Explain the tests you applied to show that \( \bar{x} \) is not a constrained minimizer.

(c) Apply Approach 1 with \text{maxit} = 15 and \text{ftol} = 1.0e-05. Let \( x_0 = (-1.2, -1.9) \) with initial \( \rho = 1 \).

At the 4th iteration of Newton’s method for each value of \( \rho \), print \( k \), \( x \), \( f \), \( c \), and \( \| \nabla P_0 \| \), using scientific notation for the latter values and showing at least 6 significant digits. For each value of \( \rho \), after your Newton iterations have terminated, print \( \| x^*(\rho) - x^* \| \) and \( \| \lambda(\rho) - \lambda^* \| \). Comment on how these differences seem to be related to the value of \( \rho \).

(d) Letting \text{maxit} = 8 and \text{ftol} = 1.0e-05, apply Approach 2, with \( x_0 = (-1.2, -1.9) \) and \( \lambda_0 = 0 \).

Comment on the behavior of the iterates. Do they appear to be converging quadratically to \( (x^*, \lambda^*) \)? Explain their behavior.
(e) Letting \textsf{maxit} = 8 and \textsf{ftol} = 1.0e-05, repeat (d) with \( x_0 = (1.2, -1.8) \) and \( \lambda_0 = 0 \). What is a noticeable difference between the results for (e) and the results in (d)? Explain the behavior of the Newton-Lagrange method by considering the figure.

\textbf{Exercise 5.4.} Augmented Lagrangian methods rely on existence of a finite penalty parameter \( \bar{\rho} \) such that the Hessian of the augmented Lagrangian,

\[
\nabla^2 L_A(x, \lambda, \rho) = H(x) - \sum_{i=1}^{m} \lambda_i H_i(x) + \rho J(x)^T J(x),
\]

is positive definite for \( \rho > \bar{\rho} \).

Consider the eigenvalues of the Hessian of the augmented Lagrangian function for the objective function and constraint of (5.3), evaluated at the optimal \( x \) and \( \lambda \).

(a) Show by computation that, when \( \rho = 0 \), then \( \nabla^2 L_A(x^*, \lambda^*, \rho) \) is not positive definite.

(b) Find a value of \( \rho \) for which \( \nabla^2 L_A(x^*, \lambda^*, \rho) \) is positive definite.

(c) Use a 1-d zero-finding method to find \( \bar{\rho} \) such that \( \nabla^2 L_A(x^*, \lambda^*, \bar{\rho}) \) is positive semidefinite and singular.