University of Mannheim
Scientific Computing, B6 26, C306, 68131 Mannheim
Dr. Andreas Sommer (ansommer@mail.uni-mannheim.de)

Nonlinear Optimization (FSS 2023)

Exercise Sheet #7

Due on 23.04.2023 (before 13:00).

1. Newton-like methods and inexact Newton methods [4 points]

Show that inexact Newton methods can be interpreted as Newton-like methods and vice-versa.

Hint: In both cases, the Newton equation $\nabla^2 f(x^k)d^k + \nabla f(x^k)$ has a residual r^k (due to inexact solution of the equation system or due to an Hessian approximation $M_k \neq \nabla^2 f(x^k)$).

2. Superlinear Convergence of the (local) Quasi-Newton Method [3 points]

Prove Theorem 10.1: Assume that the point $\tilde{x}$ satisfies second order sufficient conditions (SOC-2). Further, assume that Algorithm 10 generates a convergent sequence (x^k) with limit $\tilde{x}$ and

$$\lim_{k\to\infty} \|H_{k+1} - H_k\| = 0\,. \tag{1}$$

Then, H_k satisfies the Dennis-Moré condition and (x^k) converges superlinearly to $\tilde{x}$.

Show that

$$\|\big(H_k - \nabla^2 f(x^k)\big)d^k\| = o(\|d^k\|)\,. \tag{2}$$

Hint: Use (1) and the Quasi-Newton equation to show that (2) holds true.

3. Positive Definiteness of Broyden Class Updates [3+3+3 points]

Show Theorem 10.2: Let H_k be symmetric. Further assume that

$$(y^k)^\top s^k \neq 0 \qquad \text{und} \qquad (s^k)^\top H_k s^k \neq 0\,.$$

Then, the matrices H_{k+1}^λ with $\lambda \in \mathbb{R}$ are well defined, symmetric and satisfy the Quasi-Newton equation

$$H_{k+1}^\lambda(x^{k+1} - x^k) = \nabla f(x^{k+1}) - \nabla f(x^k)\,.$$

In addition, if H_k is positive definite and it holds true that $(y^k)^\top s^k > 0$, then H_{k+1}^λ will be positive definite for $\lambda \geq 0$.

Without proof you can use that $H_{k+1}^\lambda = H_{k+1}^{BFGS} + \lambda\big((s^k)^\top H_k s^k\big) v^k {v^k}^\top$,
with $v^k = \dfrac{y^k}{(y^k)^\top s^k} - \dfrac{H_k s^k}{(s^k)^\top H_k s^k}$.

(a) Show that H_{k+1}^λ satisfies the Quasi-Newton equation for all $\lambda \in \mathbb{R}$.
Hint: Start by calculating $(v^k)^\top s^k$

(b) Show that, if H_k is positive definite, $(y^k)^\top s^k > 0$ <u>and</u> H_{k+1}^{BFGS} is positive definite, then H_{k+1}^λ will be positive definite for all $\lambda \geq 0$, i.e.

$$w^\top H_{k+1}^\lambda w > 0 \qquad \forall w \in \mathbb{R}^n \setminus \{0\}\,.$$

(c) Show that H_{k+1}^{BFGS} will be positive definite, if H_k is positive definite and $(y^k)^\top s^k > 0$.
Hint: Due to positive definiteness of H_k, there exists a factorization $H_k = R_k^\top R_k$ with invertible matrix R_k (e.g. use Cholesky factorization).

4. Programming assignment: Globalized Newton-like Method [12 points]

Implement the globalized Newton-like method in Algorithm (8). Use a function header

```
function X = globalnewtonlike(f, gradf, Mkf, x0, e, maxit, beta, gamma, a1, a2, p)
```

Here, `f` and `gradf` are functions that return for given x the functional value $f(x)$ and the gradient $\nabla f(x)$, `Mkf` is a function that, for given x, returns a Hessian approximation M_k, and further `x0` denotes the starting point. The iteration should be terminated if $\|\nabla f(x^k)\| \leq \epsilon$ for a given tolerance `e` $= \epsilon > 0$ or if a maximum number of iterations `maxit` has been reached. Parameters `beta` and `gamma` are β and γ from the Armijo rule, and `a1`, `a2`, `p` are the parameters α_1, α_2, p from the search direction condition (7.1). The return value `X` shall contain all iterates x^k as a matrix.

Display in every iteration on a **single line**: Iteration number k, norm of gradient $\|\nabla f(x^k)\|$, function value $f(x^k)$, components of iterate x^k, step size σ_k, the *contraction rate* $\frac{\|x^{k+1}-x^k\|}{\|x^k-x^{k-1}\|}$, a mark `N` if the Newton-like direction has been chosen or `G` if the gradient direction has been chosen. Use the `fprintf`-directive `%11.5g` for displaying double-precision values.

Test your program on the two functions

(1) $f_1(x) = log(e^{x_1} + e^{-x_1}) + x_2^2$

(2) $f_2(x) = (x_1^2 + x_2 - 11)^2 + (x_1 + x_2^2 - 7)^2$ (Himmelblau's function)

using $x_a^0 = (1.0, -0.5)^\top$, $x_b^0 = (-1.2, 1.0)^\top$, $x_c^0 = (5.0, 5.0)^\top$ as initial guesses, and choose parameters `e` $= \varepsilon = 10^{-6}$, `maxiter`$=100$, $\beta = 0.5$, $\gamma = 10^{-4}$, $\alpha_1 = \alpha_2 = 10^{-6}$, and $p = 0.1$.

Choose as Hessian approximations M_k:

(a) $M_k = \nabla^2 f(x^k)$, i.e. the exact Hessian

(b) $M_k = diag\big(\nabla^2 f(x^k)\big)$, i.e. the diagonal of the exact Hessian

(c) $M_k = \nabla^2 f(x^0)$, i.e. constant approximation by initial Hessian

Compare the results. What do you observe?

On convergence, check if the respective solution is a (local) minimum (i.e. compute the Hessian and its eigenvalues).

Why are the iterates for f_1 with Hessian approximations (a) and (b) identical?

Put all files into a single zip-archive, named by the lexicographically ordered family names of all group members separated by a hyphen, and send the zip-file to ansommer@mail.uni-mannheim.de.
Please add printouts from code and output to your submissions.
Comment your code intensely. Use a complete header that describes input and output arguments and also comment the implementation where appropriate (see the examples at the course web site).
Avoid obvious inefficiencies like repeated evaluation of identical/unchanged expressions.