TU Chemnitz
C. Helmberg

Chemnitz, October 17, 2017
due on October 30, 2017

Nonlinear Optimization

Exercises 2

1. (2) Given a convex set $C \subseteq \mathbb{R}^n$. A function $f\colon C \to \mathbb{R}$ is *quasiconvex*, if $f(\alpha x_1 + (1-\alpha)x_2) \le \max\{f(x_1), f(x_2)\}$ for all $x_1, x_2 \in C$ and all $\alpha \in (0,1)$. Prove: f is quasiconvex if and only if all level sets $S_f(r) = \{x \in C\colon f(x) \le r\}$ are convex. Are all local minima global ones as well?

2. (3) Given a convex set $C \subseteq \mathbb{R}^n$. A quasiconvex function $f\colon C \to \mathbb{R}$ is *strictly quasiconvex*, if for each $x_1, x_2 \in C$ with $f(x_1) \ne f(x_2)$ we have $f(\alpha x_1 + (1-\alpha)x_2) < \max\{f(x_1), f(x_2)\}$ for all $\alpha \in (0,1)$. Prove: If f is strictly quasiconvex, any local minimum is also a global minimum and the set of minima $\operatorname{Argmin}(f)$ is convex. What is lost, if in the definition of strict quasiconvexity the requirement is dropped that f is quasiconvex?

3. (3) Given a strictly quasiconvex $f\colon [a,b] \to \mathbb{R}$ and $a \le \lambda < \mu \le b$, prove: If $f(\lambda) < f(\mu)$, then $f(\alpha) \ge f(\mu)$ for $\alpha \in [\mu, b]$, otherwise $f(\alpha) \ge f(\lambda)$ for $\alpha \in [a, \lambda]$.

4. (4) Derivative free line-search: Let a strictly quasiconvex $f\colon [a,b] \to \mathbb{R}$ be given by a 0^{th} order oracle.

   (a) Describe an algorithm with the following property: In iteration $0 \le k \le n-1$ the search interval $[a_k, b_k]$ for the minimum is reduced by exploiting 3 after evaluating the function in
   $$\lambda_k = a_k + \frac{F_{n-k-1}}{F_{n-k+1}}(b_k - a_k) \text{ and } \mu_k = a_k + \frac{F_{n-k}}{F_{n-k+1}}(b_k - a_k),$$
   where $F_0 = F_1 = 1$ and $F_{n+2} = F_{n+1} + F_n$, $n \in \mathbb{N}_0$, is the Fibonacci sequence. What is the total number of evaluations needed?

   (b) The Golden Section Method: Now use $\theta = \frac{\sqrt{5}-1}{2}$ and
   $$\lambda_k = a_k + (1-\theta)(b_k - a_k), \quad \mu_k = a_k + \theta(b_k - a_k).$$
   By how much is the search interval reduced per iteration?

5. (8) Develop and implement in matlab a line search function of the form

   `[npoint,nval,ngrad]=linesearch(@fun,cpoint,dir,maxa,cval,cgrad,c1,c2);`

   For a continuously differentiable function f it should perform an interpolation search on $f(x_c + \alpha p)$ for $\alpha \in [0, \bar{\alpha}]$ so that the Wolfe conditions are satisfied for parameters c_1 and c_2. The parameters of the routine are `fun` for f, `cpoint` for x_c, `dir` for the descent direction p, `maxa` for $\bar{\alpha}$, `cval` for $f(x_c)$ and `cgrad` for $\nabla f(x_c)$. Return values are the new point, the new fucntion value and the new gradient. All vectors/points are column vectors. For `fun` it must be possible to plug in an arbitrary function of the form

   `[val,grad]=function_name(point)`

   (type "`help feval`" in matlab)

Send `*.m` files to `helmberg@mathematik.tu-chemnitz.de`