Scientific Computing 2

Summer term 2017
Prof. Dr. Ira Neitzel
Christopher Kacwin

universitätbonn

Sheet 5

Submission on **Tuesday, 23.5.2017**.

Exercise 1. (dual problem)

For the following optimization problems, state the corresponding dual problem. Replace the interior minimization of the dual problem by appropriate constraints.

a)
$$\min_{x\in\mathbb{R}^n} c^\top x$$
with constraints
$$Ax = a,\ x \geq 0$$
for some $c \in \mathbb{R}^n$, $A \in \mathbb{R}^{m\times n}$, $a \in \mathbb{R}^m$.

b)
$$\min_{x\in\mathbb{R}^n} \frac{1}{2} x^\top Q x + c^\top x$$
with constraints
$$Ax \leq a,\ Bx = 0$$
for some $c \in \mathbb{R}^n$, $A \in \mathbb{R}^{m\times n}$, $a \in \mathbb{R}^m$, $B \in \mathbb{R}^{p\times n}$.

(6 points)

Programming exercise 1. (Nelder-Mead method)

The goal of this programming exercise is the implementation of the Nelder-Mead method for unconstrained minimization problems.
Let $f : \mathbb{R}^n \to \mathbb{R}$ be a sufficiently smooth function and consider the optimization problem
$$\min_{x\in\mathbb{R}^n} f(x)\,.$$
The idea behind the Nelder-Mead method is to consider a n-simplex
$$S = \left\{ \sum_{i=0}^{n} \lambda_i x_i \mid \sum_i \lambda_i = 1, \lambda_i \geq 0 \right\}$$
for given points $x_0, \dots, x_n$. The corner node x_m with the highest function value is then changed to a point with a smaller function value, obtaining a new simplex. If the program is not able to do so, a shrinking operation is applied to the whole simplex. The series of simplices should converge to a local solution of f.
In detail, the Nelder-Mead method proceeds as follows:

0. Choose a starting point $x_0 \in \mathbb{R}^n$ and a starting simplex size $s > 0$. The starting simplex is given by the points $x_0 = x^{(0,0)}$ and
$$x^{(0,j)} = x^{(0,0)} + s e_j, \quad j = 1 \dots, n$$
where $(e_j) \in \mathbb{R}^n$ is the standard basis. Set $k = 0$.

1. Determine the points $x^{(k,m)}$ with
$$f(x^{(k,m)}) = \max\{f(x^{(k,0)}), \dots, f(x^{(k,n)})\}$$
and $x^{(k,l)}$ with
$$f(x^{(k,l)}) = \min\{f(x^{(k,0)}), \dots, f(x^{(k,n)})\}\,.$$
If the stopping criterion
$$\left|\frac{1}{n!}\det(x^{(k,1)} - x^{(k,0)}, \dots, x^{(k,n)} - x^{(k,0)})\right| < \epsilon$$
is satisfied (the volume of the simplex is small enough), stop the algorithm and return $x^{(k,l)}$.
Otherwise, compute the barycenter of the face opposing $x^{(k,m)}$, i.e.
$$b^{(k,m)} = \frac{1}{n}\sum_{i\neq m} x^{(k,i)}\,.$$

2. Compute the reflection candidate
$$x^r = b^{(k,m)} + \gamma(b^{(k,m)} - x^{(k,m)})$$
with reflection parameter $0 < \gamma \leq 1$. We consider 3 cases:

   a)
   $$f(x^r) < f(x^{(k,l)})\,.$$
   In this case, we try to find an even better point x^e via expansion
   $$x^e = b^{(k,m)} + \beta(x^r - b^{(k,m)})$$
   with expansion parameter $\beta > 1$. We set
   $$x^{(k+1,m)} = \begin{cases} x^e & f(x^e) < f(x^r) \\ x^r & f(x^r) \leq f(x^e)\,. \end{cases}$$
   Furthermore, set $x^{(k+1,i)} = x^{(k,i)}$ for $i \neq m$, set $k = k+1$, and go to step 1.

   b)
   $$f(x^{(k,l)}) \leq f(x^r) \leq \max\{f(x^{(k,j)}) \mid j \neq m\}\,.$$
   We set $x^{(k+1,m)} = x^r$, $x^{(k+1,i)} = x^{(k,i)}$ for $i \neq m$, $k = k+1$, and go to step 1.

   c)
   $$f(x^r) > \max\{f(x^{(k,j)}) \mid j \neq m\}\,.$$
   We first try an inner or outer partial contraction. Set
   $$x^c = \begin{cases} b^{(k,m)} + \alpha(x^{(k,m)} - b^{(k,m)}) & f(x^r) \geq f(x^{(k,m)}) \\ b^{(k,m)} + \alpha(x^r - b^{(k,m)}) & f(x^r) < f(x^{(k,m)}) \end{cases}$$
   with contraction constant $0 < \alpha < 1$. If $f(x^c) < f(x^{(k,m)})$, we set $x^{(k+1,m)} = x^c$, $x^{(k+1,i)} = x^{(k,i)}$ for $i \neq m$, $k = k+1$, and go to step 1. Otherwise, all attempts to find a new simplex corner node have failed. We do a complete contraction w.r.t. $x^{(k,l)}$, i.e., we set $x^{(k+1,l)} = x^{(k,l)}$ and for $i \neq l$
   $$x^{(k+1,i)} = \frac{1}{2}(x^{(k,i)} + x^{(k,l)})\,.$$
   Set $k = k+1$ and go to step 1.

Your task is the following:
Implement the Nelder-Mead method for space dimension $n = 2$. Use the parameters $x_0 = (-1.9, 2)^\top$, $s = 1$, $\alpha = 0.5$, $\beta = 2$, $\gamma = 1$, $\epsilon = 10^{-9}$ to test your implementation on the Rosenbrock function $f(x_1, x_2) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2$. Visualize your results (for instance with gnuplot), plotting isolines of the function and all points at which function evaluations take place.

(20 points)

The programming exercise should be handed in either during the exercise classes (bring your own laptop!) or in the HRZ-CIP-Pool, after making an appointment at "kacwin@ins.uni-bonn.de". All group members need to attend the presentation of your solution. Closing date for the programming exercise is the 1.6.2017.