The Chinese University of Hong Kong, Shenzhen
SDS · School of Data Science

Andre Milzarek · Summer Semester 2020

MAT 3007 – Optimization

Exercise Sheet 7

Assignment A7.1 (Implementing the Newton Method): (approx. *30* points)
Implement the globalized Newton method that was presented in the lecture in `MATLAB` or `Python` to solve the optimization problem:

$$\min_{x\in\mathbb{R}^2} \; f(x) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2. \tag{1}$$

The pseudo-code for the full Newton method is given as follows:

Algorithm 1: The Globalized Newton Method

Initialization: Select an initial point $x^0 \in \mathbb{R}^n$ and parameter $\gamma, \gamma_1, \gamma_2, \sigma \in (0,1)$ and `tol`.
for $k = 0, 1, \ldots$ **do**
- Compute the Newton direction s^k as solution of the linear system of equations:

$$\nabla^2 f(x^k) s^k = -\nabla f(x^k).$$

- If $-\nabla f(x^k)^\top s^k \geq \gamma_1 \min\{1, \|s^k\|^{\gamma_2}\}\|s^k\|^2$, then accept the Newton direction and set $d^k = s^k$. Otherwise set $d^k = -\nabla f(x^k)$.
- Choose a step size α_k by backtracking line search and calculate $x^{k+1} = x^k + \alpha_k d^k$.
- If $\|\nabla f(x^{k+1})\| \leq$ `tol`, then STOP and x^{k+1} is the output.

The following input parameters should be considered:

- $x^0 = (2, 5)^\top$ – the initial point.
- `tol` $= 10^{-6}$ – the tolerance parameter.
- $\sigma = \frac{1}{2}$, $\gamma = 0.1$ – parameters for backtracking and the Armijo condition.
- $\gamma_1 = 10^{-6}$, $\gamma_2 = 0.1$ – parameters for the descent condition.

The method should return the final iterate x^k that satisfies the stopping criterion.

a) Implement the globalized Newton method for problem (1) as described in Algorithm 1. Test your implementation using the given parameter choices and report the number of iterations, the final objective function value, and the point to which the method converged. Did the approach always select the Newton direction?

Hint: You can use `MATLAB`'s backslash operator `x = A\b` to solve linear equations `Ax = b`.

b) Adjust your code such that the iterates x^k are saved and returned. The unique solution of this problem is $x^* = (1,1)^\top$. Plot the sequence $(\|x^k - x^*\|)_k$ (with the iteration number k as x-axis) using a logarithmic scale. Which type of convergence can be observed?

c) Run the gradient method with backtracking ($\sigma = \frac{1}{2}$, $\gamma = 0.1$) on this problem using the same initial point x^0 and tolerance $\texttt{tol} = 10^{-6}$. Compare the performance of the Newton and gradient method.

   Hint: You might need to adjust the maximum number of iterations in the gradient method to reach the desired accuracy.

Assignment A7.2 (Branch-and-Bound Method): (approx. *30* points)
Use the branch-and-bound method to solve the following integer program.

$$\begin{array}{ll} \text{maximize} & 2x + y \\ \text{subject to} & -3x + 2y \quad \leq 5 \\ & -x - 2y \quad \leq -2 \\ & 5x + 2y \quad \leq 17 \\ & x, y \in \mathbb{Z}. \end{array}$$

You are allowed to use an LP solver to solve each of the relaxed linear program. Please specify the branch-and-bound tree and what you did at each node (similar to the procedure introduced in the lecture).

Assignment A7.3 (Multiple Knapsacks): (approx. *40* points)
Suppose we have a set of n many items and a set of m different knapsacks. For each item i and knapsack j, the following information is given:

- The item i has value (preference) v_i.
- The weight of item i is a_i.
- The capacity of knapsack j is at most C_j.

a) Formulate an integer program to maximize the total value of items that can be packed in the different knapsack while adhering to the capacity constraint (i.e., the total weight of items in each bag j is not allowed to be larger than C_j).

   Hint: You can introduce variables x_{ij} to denote whether item i is placed in knapsack j.

b) Consider the following list of items and bags:

Item	Laptop	T-Shirt	Swim. Trunks	Sunglasses	Apples	Opt. Book	Water
Value	2	1	3	2	1	4	2
Weight	2	0.5	0.5	0.1	0.5	1	1.5

Knapsack 1	Knapsack 2
$C_1 = 3$	$C_2 = 2$

   Formulate the corresponding IP in that case. What are the optimal solutions to the IP and its LP relaxation (you can use `MATLAB` or `CVX` to solve the problems)? Is there an integrality gap in this case?

Sheet 7 is due on **Aug, 2nd**. Submit your solutions before **Aug, 2nd**, **11:00 am**.