MAST30013 Assignment 2 2021 LaTex

Michael Le (998211)

April 28, 2021

Question 1
Part a Solution:

Given the function f: $R^3 \rightarrow R$:

$$f(x) = x_1^4 + x_2^4 + x_3^4 + x_1^2x_2^2 - 4x_1^3 - 12x_1x_2^2 - x_1x_2x_3 + 12x_1x_3 - 4x_3 + 1$$

<u>Gradient Vector:</u>

$$\nabla \mathbf{f(x)} = \begin{bmatrix} \frac{\partial f(x)}{\partial x_1} \\ \frac{\partial f(x)}{\partial x_2} \\ \frac{\partial f(x)}{\partial x_3} \end{bmatrix} = \begin{bmatrix} 4x_1^3 + 2x_1x_2^2 - 12x_1^2 - 12x_2^2 - x_2x_3 + 12x_3 \\ 4x_2^3 + 2x_1^2x_2 - 24x_2x_1 - x_1x_3 \\ 4x_3^3 - x_1x_2 + 12x_1 - 4 \end{bmatrix}$$

<u>Hessian Matrix:</u>

$$\nabla^2 \mathbf{f(x)} = \begin{bmatrix} \frac{\partial^2 f(x)}{\partial x_1^2} & \frac{\partial^2 f(x)}{\partial x_1 \partial x_2} & \frac{\partial^2 f(x)}{\partial x_1 \partial x_3} \\ \frac{\partial^2 f(x)}{\partial x_2 \partial x_1} & \frac{\partial^2 f(x)}{\partial x_2^2} & \frac{\partial^2 f(x)}{\partial x_2 \partial x_3} \\ \frac{\partial^2 f(x)}{\partial x_3 \partial x_1} & \frac{\partial^2 f(x)}{\partial x_3 \partial x_2} & \frac{\partial^2 f(x)}{\partial x_3^2} \end{bmatrix} = \begin{bmatrix} 12x_1^2 + 2x_2^2 - 24x_1 & 4x_1x_2 - 24x_2 - x_3 & -x_2 + 12 \\ 4x_1x_2 - 24x_2 - x_3 & 12x_2^2 + 2x_1^2 - 24x_1 & -x_1 \\ -x_2 + 12 & -x_1 & 12x_3^3 \end{bmatrix}$$

Part bi Solution:

Minimiser	f-value	Number of times	Ave iterations per search	Ave time per search (sec)
(4.3234,-4.1541,-2.5437)	-373.4173	539	13	0.01477
(-1.0930,-0.0622,1.6256)	-13.2455	60	8.9	0.007232
(4.1096,3.9638,-1.9360)	-296.1545	401	18	0.009923
Newton Method				
Minimiser	f-value	Number of times	Ave iterations per search	Ave time per search (sec)
(4.3234,-4.1541,-2.5437)	-373.4173	337	7	0.00742
(-1.0930,-0.0622,1.6256)	-13.2455	29	4	0.00696
(4.1096,3.9638,-1.9360)	-296.1545	634	6	0.01763
BFGS Method				
Minimiser	f-value	Number of times	Ave iterations per search	Ave time per search (sec)
(-1.0930,-0.0622,1.6256)	-13.2455	47	5	0.00669
(4.1096,3.9638 ,-1.9361)	-296.1545	567	9.2	0.00825
(4.3235,-4.1541,-2.5438)	-373.4173	386	8.7	0.00745

Part bii Solution:

The BFGS converges a lot more slower compared to the Newton's Method and Steepest Descent Method. Which is the general descent method in which d^k is chosen as the BFGS direction. Newton's Method involves increasing the rate of convergence of an iterative descent using the Hessian function which is a classical second order method. While the Steepest Descent Method decides at each step we choose the steepest descent direction as our descent direction, and the step size which minimises the function in the steepest descent direction. Overall, we conclude that the f value is -373.417 is the global minimum with each x coordinate between the domain -10 and 10 at the point of (4.3234,-4.1541,-2.5437).

END OF ASSIGNMENT

Appendix MATLAB script.m

```
% script to call steepestDescentMethod, NewtonMethod, BFGS (2D)

tStart0 = tic; % starts a stopwatch timer to measure performance
numInstances = 1000;

for i = 1:numInstances
    tStart = tic;  % internal timer
    x0 = -10 + 20*rand(1,3);
    tolerance1 = 0.01; %tolerance for search methods
    tolerance2 = 0.00001; % tolerance for Golden section search
    T = 10; %step size
    disp('Initial condition: ');
    disp(x0);

    [xmin,f,k] = steepestDescentMethod('f', 'gradf', x0, tolerance1, tolerance2, T) ;

    disp('Steepest Descent Method');
    display([xmin, f, k]);

    [xmin,f,k] = NewtonMethod('f', 'gradf', 'hessf', x0, tolerance1, tolerance2, T);

    disp('Netwon Method');
    disp([xmin, f, k]);

    [xmin,f,k] = BFGS('f', 'gradf', x0, eye(3), tolerance1, tolerance2, T) ;

    disp('BFGS');
    disp([xmin, f, k]);

    tElapsed = toc(tStart);
    disp(['Time used is ', num2str(tElapsed)]);
end

averageTime = toc(tStart0)/numInstances;
disp(['Average time for ', num2str(numInstances), ' is ', num2str(averageTime)]);
```