Exercise 9 Sthavishtha Bhopalam Rajakumar December 26, 2017

1 Introduction

The objective of the current exercise is to get acquainted with Generalized Newton method (also called as Newton-Raphson) and the Secant Method (for calculating Jacobian) to compute the maximum of a two-variable exponential function.

2 Algorithm Description

For the two-variable function considered :

$$F(x, y) \ = \ e^{-(x \ -x_o)^2 - (y \ -y_o)^2} \tag{1}$$

To compute the maximum, its trivial to find the gradients of the function $F(x, y)$.
The system of equations can be formulated as:

$$\vec{f} \ = \ (f_x \ , \ f_y) \tag{2}$$

where,

$$f_x \ = \ -2 \ e^{-(x \ -x_o)^2 - (y \ -y_o)^2} \ (x \ - \ x_o) \tag{3}$$

$$f_y \ = \ -2 \ e^{-(x \ -x_o)^2 - (y \ -y_o)^2} \ (y \ - \ y_o) \tag{4}$$

The Jacobian can now be formulated as :

$$J = \begin{Bmatrix} \frac{\partial f_x}{\partial x} & \frac{\partial f_x}{\partial y} \\ \frac{\partial f_y}{\partial x} & \frac{\partial f_y}{\partial y} \end{Bmatrix}$$

After substituting the necessary equations, we can simplify the Jacobian (if using the Analytical formula) as :

$$J = e^{-(x \ -x_o)^2 - (y \ -y_o)^2} \begin{Bmatrix} -2 \ + \ 4 \times (x \ - \ x_o)^2 & 4 \times (x \ - \ x_o) \ (y \ - \ y_o) \\ 4 \times (x \ - \ x_o) \ (y \ - \ y_o) & -2 \ + \ 4 \times (y \ - \ y_o)^2 \end{Bmatrix}$$

The two-dimensional Jacobian matrix can now be inverted by easily, as done for any 2×2 matrix A of the form:

$$A = \begin{Bmatrix} a & b \\ c & d \end{Bmatrix}$$

$$A^{-1} = \frac{1}{ad \ - \ bc} \begin{Bmatrix} d & -b \\ -c & a \end{Bmatrix}$$

The Generalized Newton Method can now be used as (where $n+1$ and n are the next and current time step levels respectively):

$$\vec{x}^{n+1} \ = \ \vec{x}^{n} \ - \ J^{-1} \ f(\vec{x}^{n}) \tag{5}$$

On using the Secant method, the Jacobian can be evaluated as :

$$J_{i,j}(\vec{x}) \ = \ \frac{f_i(\vec{x} \ + \ h_j \vec{e_j}) \ - \ f_i(\vec{x})}{h_j} \tag{6}$$

where h_j is approximately taken to be $x_j \ \times \sqrt{\epsilon}$, with ϵ being the computer precision (taken as 10^{-16} in the current work).

3 Results and Discussions

3.1 Task 1

For this case, x_o and y_o were taken to be 3 and 4 for simplicity. Using several guess values of x and y, it was found that the respective values exceeding 2.6 and 3.7 could lead to converged solutions, close to the true solutions in a few iterations itself. This verifies the fact that Generalized Newton Methods require the initial guess values to lie close to the exact solution (the guess can be iterated till stability is obtained), an inherent disadvantage of this method. It also demonstrates second order accuracy, which substantiates its fast convergence (in a few iterations itself). Notice Figures 1, 2 to observe the convergence of x and y, within a few iterations.

To overcome the instability arising due to the guess values, an alternative is to incorporate a Relaxation factor λ (varying from 0 to 1) as follows in the Modified Generalized Newton Equation as :

$$\vec{x}^{n+1} = \vec{x}^{n} - \lambda \, J^{-1} \, f(\vec{x}^{n}) \tag{7}$$

For this case, it was found that a maximum λ value close to 0.72 was necessary to achieve stable approximate solutions. Mathematically, incorporation of this relaxation factor reduces the successive oscillations of the error and subsequent divergence from the true solution (as seen in Figures 3 and 4).

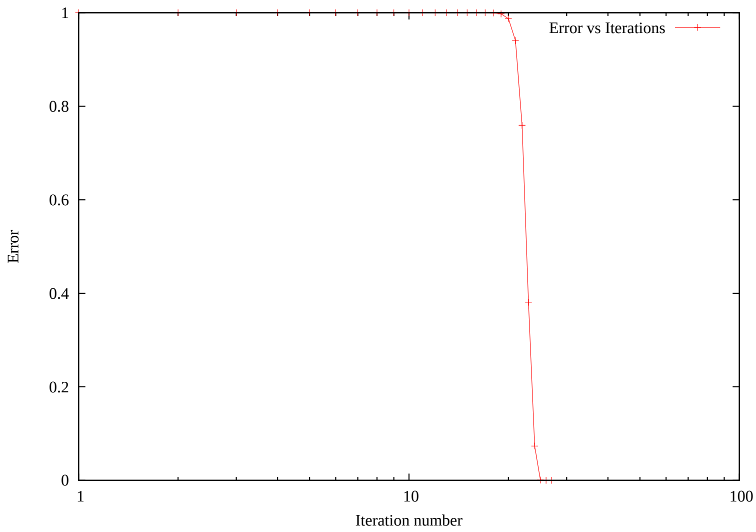

Figure 1: Graph of the error obtained versus the iterations

3.2 Task 2

Additionally, Secant method was also used to compute the numerical solutions, wherein a numerical approximation of the Jacobian was incorporated as seen in Equation 6. Though this has a lower convergence rate than its counterpart Newton-Raphson method (described earlier), the Figure 5 may be misleading in this sense as the former requires much fewer iterations than the latter. To clear the confusion, theory states the *convergence rate* as the deciding phenomenon, not the number of iterations. This may not be clearly visible in this case, as the initial error in the case of Newton-Raphson method remains close to unity for a relative large number of iterations. Fitting the two plots (using the gnuplot `fit` tool) revealed the convergence rate of Newton-Raphson to be higher than the Secant method (consistent with the theory).

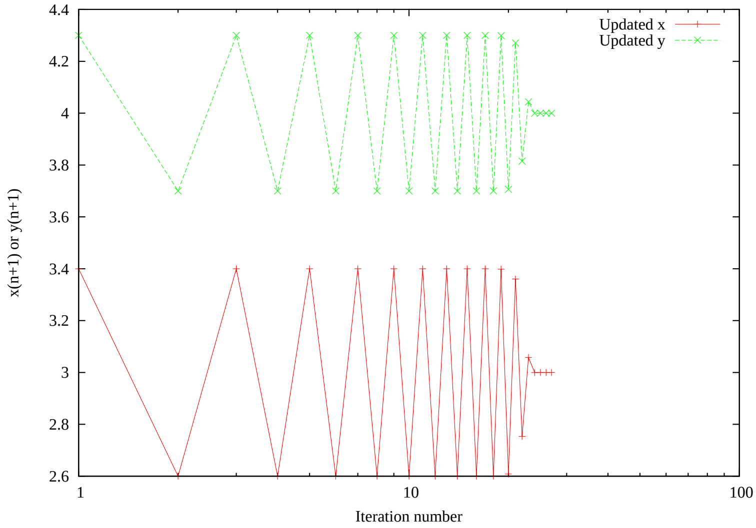

Figure 2: Graph of the new values of x and y obtained versus the iterations

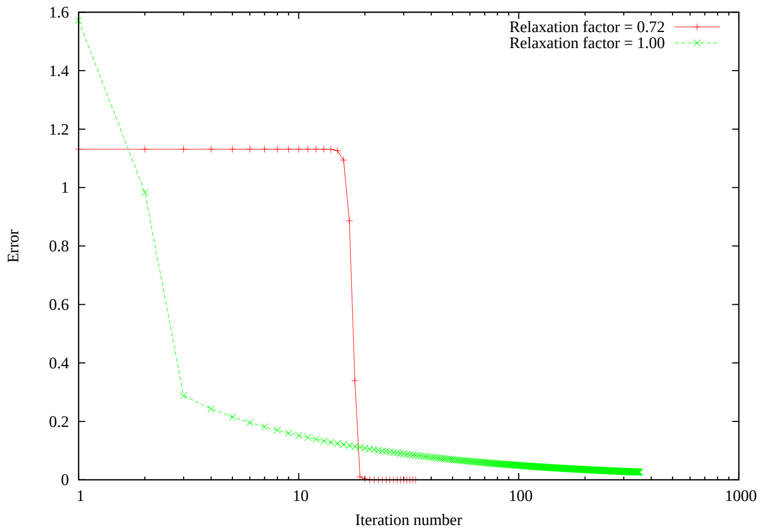

Figure 3: Graph of the error captured on incorporating the relaxation factor, for particular guess values (Notice the fewer number of iterations required on incorporating the relaxation factor to reach convergence)

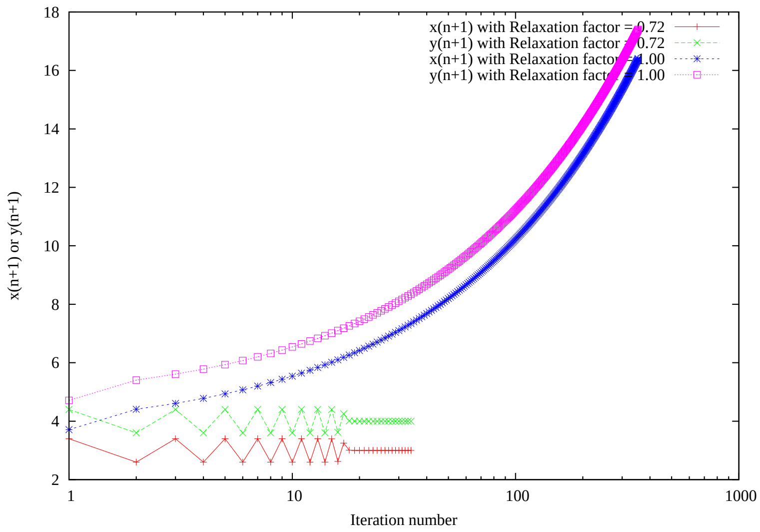

Figure 4: Graph of the error obtained for guess values of x and y taken to be 2.6 and 3.7 respectively. (Using the relaxation factor of 1.00 causes divergence)

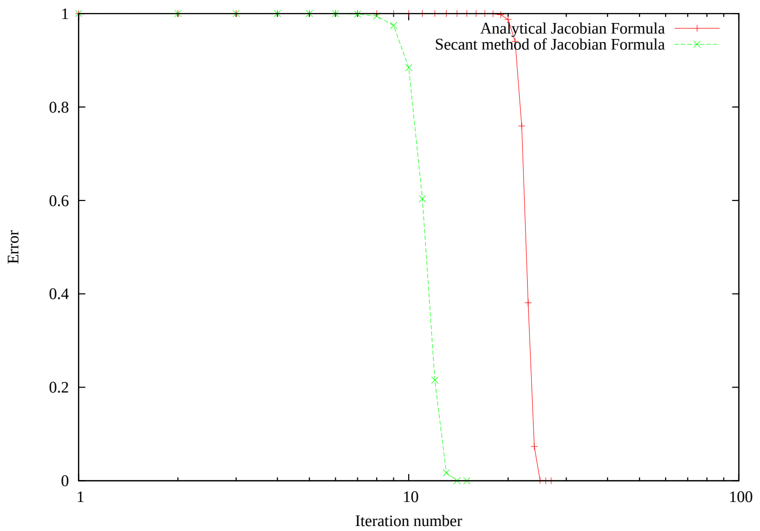

Figure 5: Comparison of the secant and analytical method of calculating the Jacobian