Seth Strayer
Project 3
11/28/18
Dr. Neilan

1. Introduction

In this report, we wish to apply Newton's method to several functions and examine the behavior of the iterates. I.e., we wish to examine if the iterates converge to the given root, and if so, calculate their rate of convergence. The functions we would like to examine are given by:

$$f(x) = \begin{cases} \sqrt{x} & x \geq 0 \\ -\sqrt{-x} & x < 0 \end{cases} \tag{1}$$

$$f(x) = \begin{cases} \sqrt[3]{x^2} & x \geq 0 \\ -\sqrt[3]{x^2} & x < 0 \end{cases} \tag{2}$$

Note that in either case, the root $\alpha = 0$. Furthermore, for $p \in (0, \infty)$, define

$$f(x) = \begin{cases} |x|^p & x \geq 0 \\ -|x|^p & x < 0 \end{cases} \tag{3}$$

We would like to find all values $p \in (0, \infty)$ such that the sequence $(x_n)_{n=1}^{\infty}$ generated by Newton's method converges to 0 for any initial guess x_0. Note that throughout the report, we assume that the initial guess x_0 is sufficiently close to the root α, as defined by Newton's method.

2. Solution

By first solving the problem presented by equation (3), we can easily verify the results we obtain for the application of Newton's method to equations (1) and (2). For this problem, we would like to find all values $p \in (0, \infty)$ such that the sequence $(x_n)_{n=1}^{\infty}$ generated by Newton's method converges to the root $\alpha = 0$ for any initial guess x_0. To find the desired values of p, we can used fixed point iteration in conjunction with Newton's Method. By definition of fixed point iteration, we declare

$$x_{(n+1)} = g(x_n) \tag{4}$$

where, by definition of Newton's method,

$$g(x_n) = x_n - \frac{f(x_n)}{f'(x_n)} \tag{5}$$

The iterates will converge to the root α if

$$|g'(\alpha)| < 1 \tag{6}$$

From equation (5), we have

$$|g'(x)| = \frac{|f(x)||f''(x)|}{|f'(x)|^2} \tag{7}$$

Note that the indices have been dropped for simplicity. Also note that with the function provided by equation (3), along with equation (7) as defined above, we can write that $|f(x)| = |x^p|$ for all x. Then, for any x, we have that $|f(x)| = |x^p|$, $|f'(x)| = |px^{(p-1)}||$, and $|f''(x)| = |p(p-1)x^{(p-2)}|$. Substituting into equation (7), we have:

$$|g'(x)| = \frac{|p(p-1)x^p x^{(p-2)}|}{|px^{(p-1)}|^2}$$

$$|g'(x)| = \frac{|p(p-1)x^{(2p-2)}|}{|p^2 x^{(2p-2)}|}$$

$$|g'(x)| = \frac{|p(p-1)|}{|p^2|}$$

Finally,

$$|g'(x)| = \frac{|(p-1)|}{|p|}$$

From definition (6), we wish to find p such that

$$\begin{gathered} -1 < \frac{p-1}{p} < 1 \\ -p < p-1 < p \\ -2p < -1 < 0 \\ p > \frac{1}{2} > 0 \end{gathered}$$

Thus, for any initial guess x_0, the sequence $(x_n)_{n=1}^{\infty}$ generated by Newton's method converges to 0 for values of $p > \frac{1}{2}$. Let us summarize this result here:

$$p > \frac{1}{2} \tag{8}$$

From the result given in (8), we hypothesize that Newton's method will converge for the function given by equation (2) ($p = \frac{2}{3}$) but will not converge for the function given by equation (1) ($p = \frac{1}{2}$). Allow us to now investigate this hypothesis.

The script files used to apply Newton's method to the functions given by equations (1) and (2) are given by *newton_1.m* and *newton_2.m*, respectively. The main script used to run these m-files is given by *Project_3.m*. This script file takes user input in the form of the initial guess, x_0, the error bound, and the maximum number of iterates, say, n.

The error bound and maximum number of iterates will be fixed at $1E-10$ and $n = 35$, respectively. Considering these inputs and the assumption that the initial guess is sufficiently close to the root $\alpha = 0$, we should be able to easily detect the convergence/divergence of Newton's method. I.e., these inputs should allow a root to be calculated, provided convergence.

Let us first consider the function given by (1). We need to consider two initial guesses (one negative and one positive). First choose an initial guess of $x_0 = 0.25 \geq 0$. The output of *newton_1.m* for $x_0 = 0.25$ is shown in figure 1, below.

```
iteration =

   2.2000e+01   2.5000e-01   5.0000e-01   1.0000e+00   5.0000e-01


 it_count x f(x) df(x) error

iteration =

   2.3000e+01  -2.5000e-01  -5.0000e-01   1.0000e+00  -5.0000e-01


 it_count x f(x) df(x) error

iteration =

   2.4000e+01   2.5000e-01   5.0000e-01   1.0000e+00   5.0000e-01


 it_count x f(x) df(x) error

iteration =

   2.5000e+01  -2.5000e-01  -5.0000e-01   1.0000e+00  -5.0000e-01

The number of iterates calculated exceeded max_iterate.
An accurate root was not calculated.
Output argument "root" (and maybe others) not assigned during call to "newton_1".

Error in Project_3 (line 9)
root = newton_1(x0,error_bound,max_iterate);
```

FIGURE 1. Output of Newton's Method for Equation (1) with $x_0 = 0.25$

Without performing any analysis, we can easily see that the iterates $(x_n)_{n=1}^{\infty}$ do not converge. In particular, the values oscillate from $[0.25, -0.25]$, for which $f(x)$ oscillates from $[0.5, -0.5]$. By observation, the error in the iterates, ϵ, given by

$$\epsilon = x_{n+1} - x_n \tag{9}$$

also oscialltes from $[0.5, -0.5]$. Thus, Newton's method does not converge for the function given by equation (1) with an initial guess $x_0 = 0.25 \geq 0$.

Let us now consider an initial guess of $x_0 = -0.25 < 0$. The output of *newton_1.m* for $x_0 = -0.25$ is shown in figure 2 on page 4.

We note the exact same behavior in this case, only with a change in values due to the negative sign in front of the radical as provided in equation (1). Hence, Newton's method *does not* converge for the function provided by equation (1) and *any* initial guess x_0.

```
iteration =

   2.2000e+01  -2.5000e-01  -5.0000e-01   1.0000e+00  -5.0000e-01


 it_count x f(x) df(x) error

iteration =

   2.3000e+01   2.5000e-01   5.0000e-01   1.0000e+00   5.0000e-01


 it_count x f(x) df(x) error

iteration =

   2.4000e+01  -2.5000e-01  -5.0000e-01   1.0000e+00  -5.0000e-01


 it_count x f(x) df(x) error

iteration =

   2.5000e+01   2.5000e-01   5.0000e-01   1.0000e+00   5.0000e-01

The number of iterates calculated exceeded max_iterate.
An accurate root was not calculated.
Output argument "root" (and maybe others) not assigned during call to "newton_1".

Error in Project_3 (line 9)
root = newton_1(x0,error_bound,max_iterate);
```

FIGURE 2. Output of Newton's Method for Equation (1) with $x_0 = -0.25$

Finally, let us consider the case for the function given by equation (2). We will keep the error bound and maximum number of iterates the same as before. First choose an initial guess of $x_0 = 0.25 \geq 0$. The output of *newton_2.m* for $x_0 = 0.25$ is shown in figure 3 on page 5.

We note that, indeed, Newton's method converges for this trial. In particular, after 32 iterations, the script calculates a root of $r = -2.910383045673369e - 11$. We note that this value is very close to the actual root of $\alpha = 0$. We are primarily interested with the rate of convergence. Computationally, we can calculate this rate, say β by calculating the ratio

$$\beta = \frac{|\epsilon_n|}{|\epsilon_{n+1}|} \tag{10}$$

Where ϵ is the error as defined by equation (9). Take an arbitrary value for n, say $n = 30$. Then from figure 3, $|\epsilon_n| = 6.9849e - 10$ and $|\epsilon_{n+1}| = 3.4925e - 10$. From equation (10), we have

$$\beta = \frac{6.9849e - 10}{3.4925e - 10} \approx 2 \tag{11}$$

We can show (proof omitted) that $\beta = 2$ for all n. Thus, the error is decreasing by a factor of two for each iterate, and we have linear convergence. This result calls for more investigation. By definition, Newton's method should display quadratic convergence (proof omitted).

```
   2.7000e+01  -1.8626e-09  -1.5139e-06   5.4183e+02  -5.5879e-09


iteration =

   2.8000e+01   9.3132e-10   9.5367e-07   6.8267e+02   2.7940e-09


iteration =

   2.9000e+01  -4.6566e-10  -6.0078e-07   8.6011e+02  -1.3970e-09


iteration =

   3.0000e+01   2.3283e-10   3.7847e-07   1.0837e+03   6.9849e-10


iteration =

   3.1000e+01  -1.1642e-10  -2.3842e-07   1.3653e+03  -3.4925e-10


iteration =

   3.2000e+01   5.8208e-11   1.5019e-07   1.7202e+03   1.7462e-10


root =

    -2.910383045673369e-11
```

FIGURE 3. Output of Newton's Method for Equation (2) with $x_0 = 0.25$

Thus, seemingly by consequence of the given function, the computational results do not completely align with the theoretical results. We make note of this issue but will not investigate it any further.

Next consider the case where the initial guess $x_0 = -0.25 < 0$. The output of *newton_2.m* for $x_0 = -0.25$ is shown in figure 4 on page 6.

We note the exact same behavior in this case, only with a change in values due to the negative sign in front of the radical as provided in equation (2). Hence, Newton's method *converges* (and does so at a linear rate) for the function provided by equation (2) and *any* initial guess x_0.

Allow us to summarize our results in the conclusion section on the following page.

```
   2.7000e+01   1.8626e-09   1.5139e-06   5.4183e+02   5.5879e-09


iteration =

   2.8000e+01  -9.3132e-10  -9.5367e-07   6.8267e+02  -2.7940e-09


iteration =

   2.9000e+01   4.6566e-10   6.0078e-07   8.6011e+02   1.3970e-09


iteration =

   3.0000e+01  -2.3283e-10  -3.7847e-07   1.0837e+03  -6.9849e-10


iteration =

   3.1000e+01   1.1642e-10   2.3842e-07   1.3653e+03   3.4925e-10


iteration =

   3.2000e+01  -5.8208e-11  -1.5019e-07   1.7202e+03  -1.7462e-10


root =

     2.910383045673369e-11
```

FIGURE 4. Output of Newton's Method for Equation (2) with $x_0 = -0.25$

3. CONCLUSION

In this report, we first discovered that for the function given by equation (3), the sequence $(x_n)_{n=1}^{\infty}$ generated by Newton's method converges to the root $\alpha = 0$ for values of $p > \frac{1}{2}$ (provided that the initial guess x_0 is sufficiently close to the root α). From this result, we hypothesized that the iterates generated by Newton's method would converge to the root α for the function provided in equation (2) ($p = \frac{2}{3}$) but will not converge for the function given by equation (1) ($p = \frac{1}{2}$). We then verified this result computationally with the use of the script files *newton_1.m* and *newton_2.m*. I.e., through these script files, we indeed found that Newton's method *does not* converge for the function provided by equation (1), yet *converges* for the function provided by equation (2). For the latter of these two functions, we calculated a linear rate of convergence. This result disagrees with the quadratic rate of convergence as theoretically anticipated by Newton's method and should be further investigated. Overall, we were able to apply the theory of Newton's method to several functions and are satisfied with our results.