APPM 4600
Homework 4

February 14, 2025

Jude Gogolewski

---

1 Cold Snap tests

We aim to learn about our system and particularly at what depth our water mains should lye to avoid an Immediate freeze based on the following equations:

$$\frac{T(x,t) - T_s}{T_i - T_s} = erf(\frac{x}{2\sqrt{\alpha t}}) \tag{1}$$

$$erf(t) = \frac{2}{\sqrt{\pi}} \int_0^t e^{-s^2} ds \tag{2}$$

with constants:
$T_s = -15^0$ constant temp during a cold snap
$T_i = 20^0$ initial soil temp
$\alpha = .138 * 10^{-6}$ thermal conductivity

(a) Formulating a root-finding problem to find $f(x) = 0$:
Let's start by inserting known values into equation (1):

$$\frac{0 - (-15)}{20 - (-15)} = erf(\frac{x}{2\sqrt{.138 * 10^{-6}(60hrs)}}) \tag{3}$$

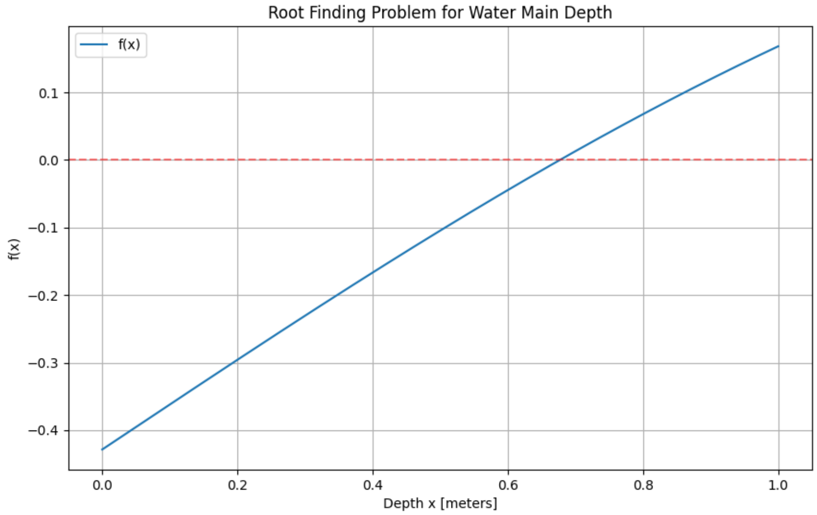
$$f(x) = erf(\frac{x}{1.6916}) - .42857 = 0 \tag{4}$$

$$f'(x) = \frac{1}{\sqrt{2\pi * 1.6916}} exp(-(\frac{x}{1.6916})^2) \tag{5}$$

solving for x, we get that the depth needed is $x \approx .67$, which we will solve more accurately later.

To find our bounds where $f(x) > 0$ we simply need an interval where $erf(\frac{x}{2\sqrt{\alpha t}}) \geq .42857$ which happens to be $[0, 1]$.

After finding our roots through two different methods, we get:



(b) Root through Bisection: 0.6769593408567403

(c) Root through Newtons Method: 0.676959340856748

2 Multiplicity

let $f(x)$ be a function with root α and multiplicity m:

(a) α has root multiplicity m when all order derivatives less than m evaluated at the point α are equal to zero:

$$f^{(k)}(a) = 0 | k = 0, 1..., m-1, f^{(m)}(a) \neq 0 \tag{6}$$

(b) based on the setup of our problem, we can write $f(x) = (x-\alpha)^m q(x)$
For the Newtons method, we define:

$$g(x) = x - \frac{f(x)}{f'(x)} = x - \frac{(x-\alpha)^m q(x)}{m(x-\alpha)^{m-1}q(x) + (x-\alpha)^m q'(x)} \tag{7}$$

To Determine the rate and order of convergence, we look at the first non-zero derivative of $g(x)$:

$$g'(\alpha) = 1 - \frac{q(\alpha)}{mq(\alpha)} = 1 - \frac{1}{m} \tag{8}$$

we can now say the if $m = 1$ we converge at least quadratically, if $m > 1$ we converge linearly at a rate $1 - \frac{1}{m}$.

(c) To show that our function $g(x)$ is second-order convergent, we need to Taylor expand our $g(x)$ and relate it to our error:

Let $x = \alpha + e$

$$e_{n+1} = g'(\alpha)e_n + \frac{g''(\alpha)}{2!}(e_n)^2 + ... \quad (9)$$

$$e_{n+1} = Ke_n^2 \quad (10)$$

therefore, we can say the error in the next step is proportional to the error in the previous step squared. In turn, the Fixed point iteration for the Newton method converges Quadratically.

(d) Part C implies that if a root has multiplicity greater than one, the Fixed point iteration will converge linearly.

3 Convergence of a Sequence

To start the definition of convergence of a sequence:

$$lim_{k\to\infty}\frac{|x_{k+1}-\alpha|}{|x_k-\alpha|^p} = \lambda \quad (11)$$

Where λ is the rate of convergence and α is the order of convergence.

$$ln(|x_{k+1}-\alpha|) = ln(\lambda|x_k-\alpha|^p) \quad (12)$$

$$ln(|x_{k+1}-\alpha|) = pln(|x_k-\alpha|) + ln(\lambda) \quad (13)$$

From this, we see that p becomes our slope and λ becomes our intercept

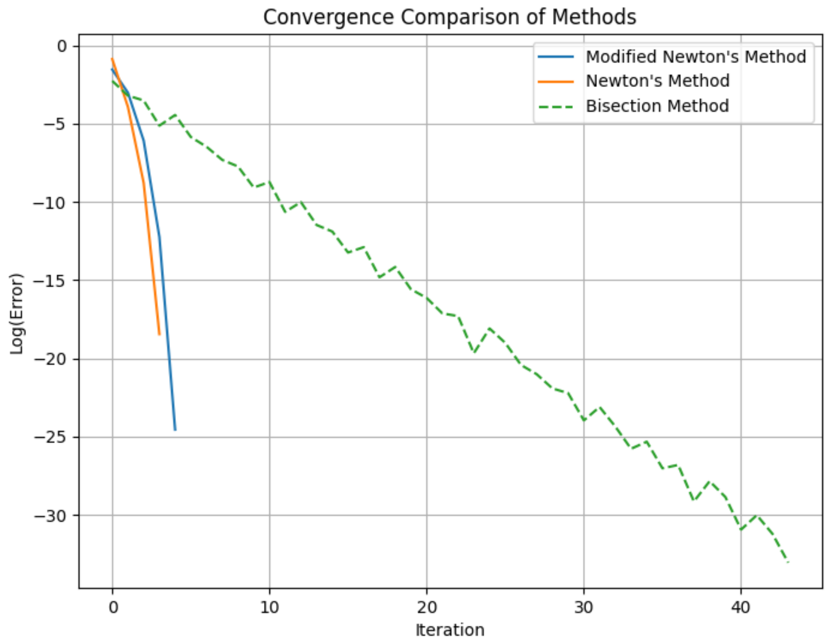
4 Roots of Higher Multiplicity

When faced with a root of multiplicity greater than 1, the Newton method begins to fail and returns to a simply linearly convergent method. To counteract this, we can manipulate Newton's method to a function to regain a quadratic or at least superlinear order of convergence.

$$f(x) = e^{3x} - 27x^6 + 27x^4e^x - 9x^2e^{2x} = (e^x - 3x^2)^3; x \in [3,5] \quad (14)$$

$$f'(x) = 3(e^x - 3x^2)^2(e^x - 6x) \quad (15)$$

Our function $f(x)$ has a multiplicity $\alpha = 3$; this tells us the regular Newton method will not be as effective as we would hope, and, in turn, modifications are needed.



(i) For regular Newton's method, we can use the formula for rate of convergence $\lambda = 1 - \frac{1}{\alpha} = 1 - \frac{1}{3} = \frac{2}{3}$

(ii) Modified Newtons method constitutes setting $g(x) = \frac{f(x)}{f'(x)}$ this method gets rid of the higher multiplicity of the root and, in turn, regains our quadratic order of convergence:

$$g(x) = \frac{(e^x - 3x^2)^3}{3(e^x - 3x^2)^2(e^x - 6x)} = \frac{(e^x - 3x^2)}{3(e^x - 6x)} \tag{16}$$

We would now define a new function to which we apply fixed point iteration again:

$$g_2(x) = x - \frac{g(x)}{g'(x)} \tag{17}$$

(iii) Here, We apply Fixed point iteration to $x_{n+1} = x - m\frac{f(x)}{f'(x)}$ where in this format $m = \alpha$ or the order of convergence, here we still gain a quadratic convergence.

$$g(x) = x - 3\frac{f(x)}{f'(x)} \tag{18}$$

Method (iii) is my preferred method as it only needs $f(x)$ and $f'(x)$ but still gains at least superlinear convergence due to the addition of the m term. Below are the three methods plotted to compare the order and convergence rate.

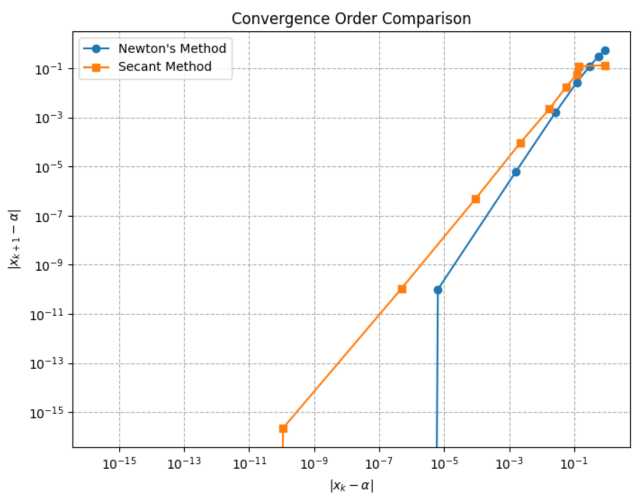
5 Newton & Secant method

Here, we aim to analyze the differences between Newton & secant's methods for root finding. For this problem, let our function take the form:

$$f(x) = x^6 - x - 1 \tag{19}$$

Both methods require one or two initial guess points we will call these points: $x_0 = 2$ & $x_1 = 1$

After Implementing Both Methods into Python and creating a Table and graph of the two methods, we can see below how these methods compare:



	Iteration	Newton Error	Secant Error
0	0	8.652759e-01	8.652759e-01
1	1	5.459041e-01	1.347241e-01
2	2	2.960148e-01	1.185951e-01
3	3	1.202468e-01	5.585363e-02
4	4	2.681429e-02	1.706831e-02
5	5	1.629136e-03	2.192588e-03
6	6	6.389942e-06	9.266960e-05
7	7	9.870171e-11	4.924528e-07
8	8	0.000000e+00	1.103035e-10
9	9	NaN	2.220446e-16
10	10	NaN	0.000000e+00

As we can see from the graph, the difference in slope of the two methods is not drastic, but it is noticeable. We can see Newton's method converges faster and in two fewer iterations. Nevertheless, the Secant method converges super linearly at a rate greater than one. The slopes of the lines in our graph have a magnitude of 2 and 1.618, which, because we are taking the log of both sides, the slope directly relates to the order of convergence.