Group Project 1 Group 2A

Group Members:

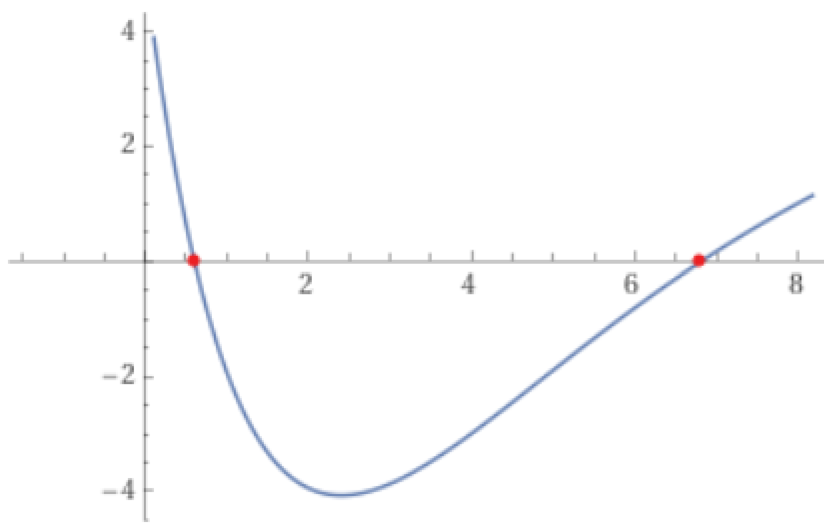
- Zun Cao - Scribe
- Aalim Abdullah
- Anjie Liu
- Anusha Singhal - Coordinator

Project Answers:

Part A:

1. $f_1(x) = 5 - 20(e^{-0.2x} - e^{-0.75x}) = 0$

   This will give the distance, x, at which the oxygen level falls to 5 mg/L.

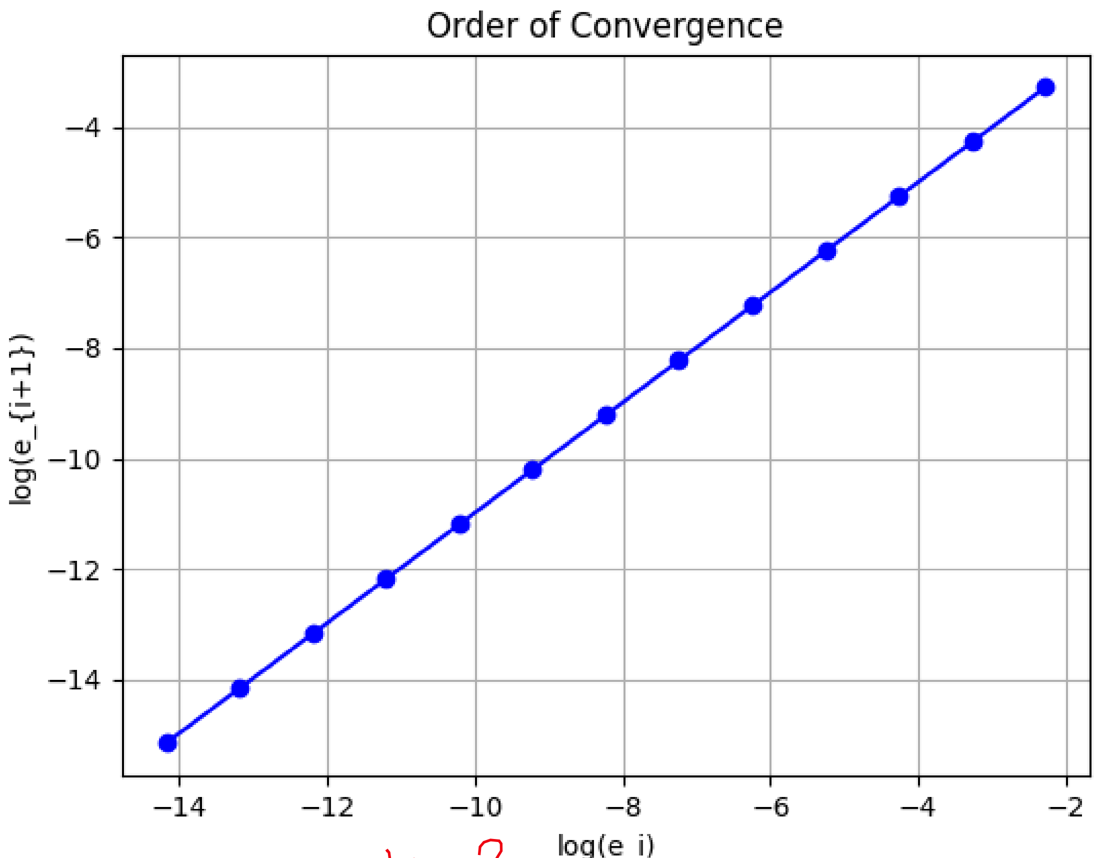


2.
3. The first root of 0.602355 along the x-axis. This is where oxygen first falls to 5 mg/L.
4. i. $g_1(x) = -\frac{4}{3}ln(e^{-0.2x} - \frac{1}{4})$ Why?

   ii. Initial approximation is 0.5. Tolerance is 0.5e-6. The solution is 0.602355.

   iii. Iteration errors = [1.02355464e-01, 3.78644098e-02, 1.40422361e-02, 5.21249458e-03, 1.93555399e-03, 7.18821110e-04, 2.66966707e-04, 9.91519119e-05, 3.68254399e-05, 1.36771578e-05, 5.07977138e-06, 1.88665557e-06, 7.00714544e-07, 2.60249355e-07]

   iv. slopes = [0.99748791386845, 0.9990615786462317, 0.9996507161185588, 0.9998701706089467, 0.9999517663637171, 0.9999820837851467, 0.999993345544567, 0.9999975284553999, 0.9999990820319946, 0.9999996590795894, 0.9999998732073845, 0.9999999530935259]



Why?

v. The theoretical slope is 1. This matches my numerics.

5. I.

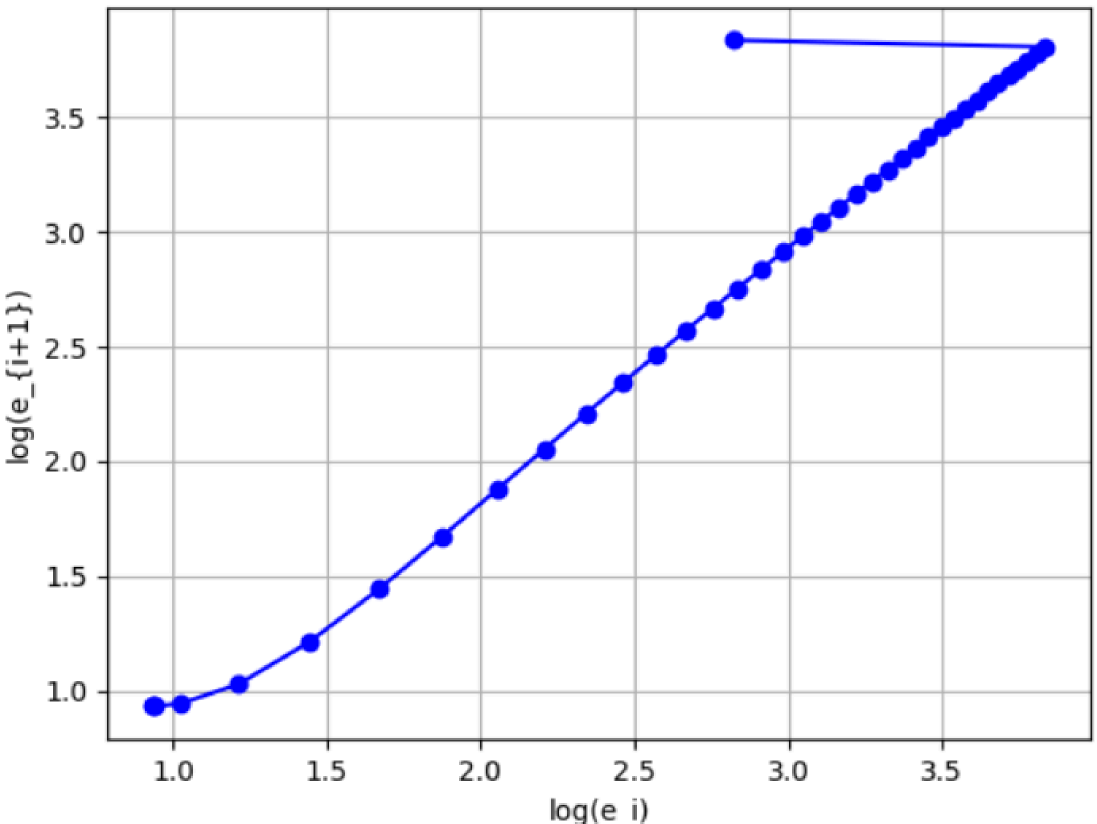
   a. The initial approximation is 20. Tolerance is 0.5e-6. Solution is r = 0.602355

Ii.

errors = [16.85840735 46.39332525 45.05999191 43.72665858 42.39332525 41.05999191
39.72665858 38.39332525 37.05999192 35.72665859 34.39332528 33.05999198
31.72665871 30.39332553 29.05999249 27.72665979 26.39332777 25.05999715
23.72666948 22.39334792 21.06003905 19.72675654 18.39352872 17.06041421
15.72753416 14.39513825 13.06373935 11.73438587 10.40920672 9.09248271
7.79269735 6.5262924 5.32403641 4.24027606 3.36125044 2.79237835
2.56958705 2.53972437 2.53923732 2.53923719]

Iii.

slopes = [-0.02880634677808675, 1.03003899573557, 1.0309694261670084, 1.0319593470632242, 1.0330146534278473, 1.0341420458322232, 1.035349172871539, 1.0366448049248533, 1.0380390475100216, 1.0395436050988307, 1.0411721097232423, 1.04294053340162, 1.0448677097687558, 1.046975998854607, 1.0492921403297963, 1.0518483552105757, 1.0546837738600612, 1.0578462871035232, 1.061394930493096, 1.0654029006641879, 1.0699612221573163, 1.0751828284979705, 1.0812061580243377, 1.0881957756290355, 1.0963338880903808, 1.1057884708009187, 1.1166257081942492, 1.1285946560301106, 1.1406232046795077, 1.1496627534753725, 1.1480659505532813, 1.117854825929891, 1.0207024761094794, 0.7981366745073677, 0.4484347120856944, 0.14058758206364105, 0.016407089730401923, 0.0002618781806765]



NOT theory

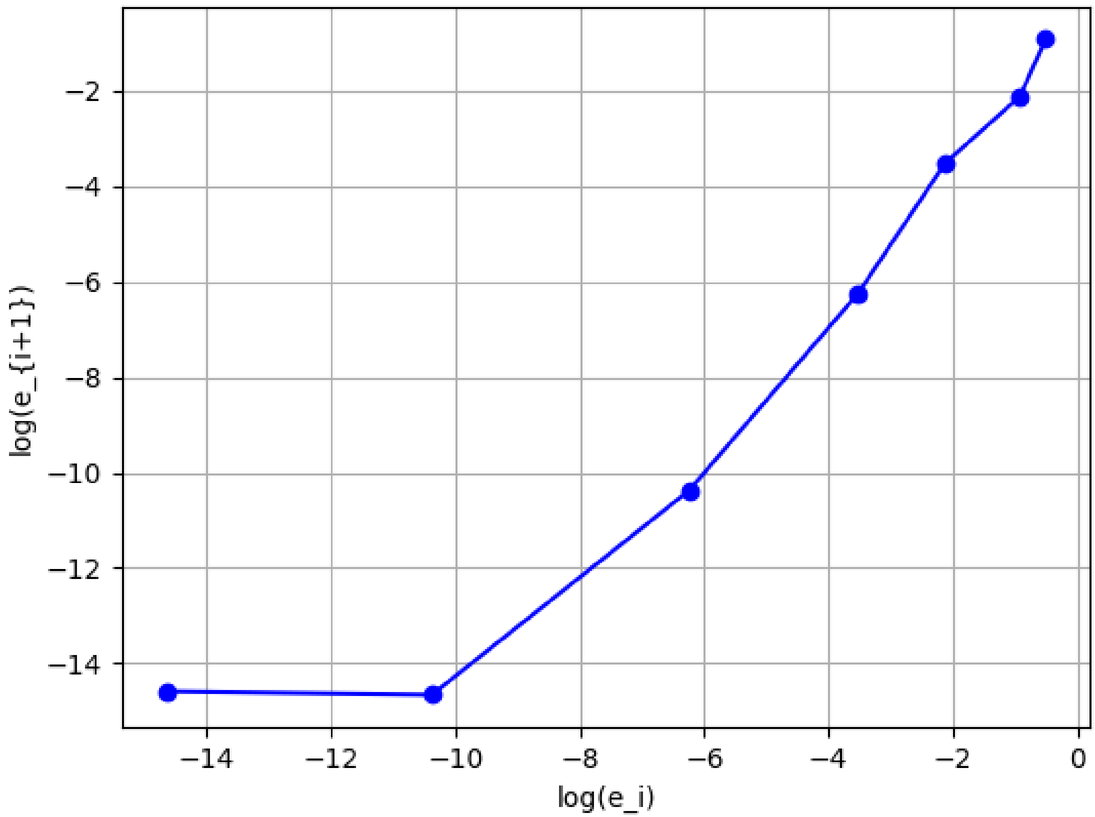
does not, slopes ~1

Iv. Theoretical slope is 0. 093110. Numerical results partially match theory.

6. i. Initial approximations = 0 and 1. Tolerance = 1e-6. Solution r = 0.602355

ii.errors = [6.02355000e-01 3.97645000e-01 1.19428602e-01 2.95769135e-02 1.94478389e-03 3.11466105e-05 4.32283129e-07 4.64364792e-07]

iii.slopes = [2.8964065548583515, 1.1603567639246903, 1.9501291496019482, 1.5188974304279332, 1.0346328685390078, -0.016736803523302732]

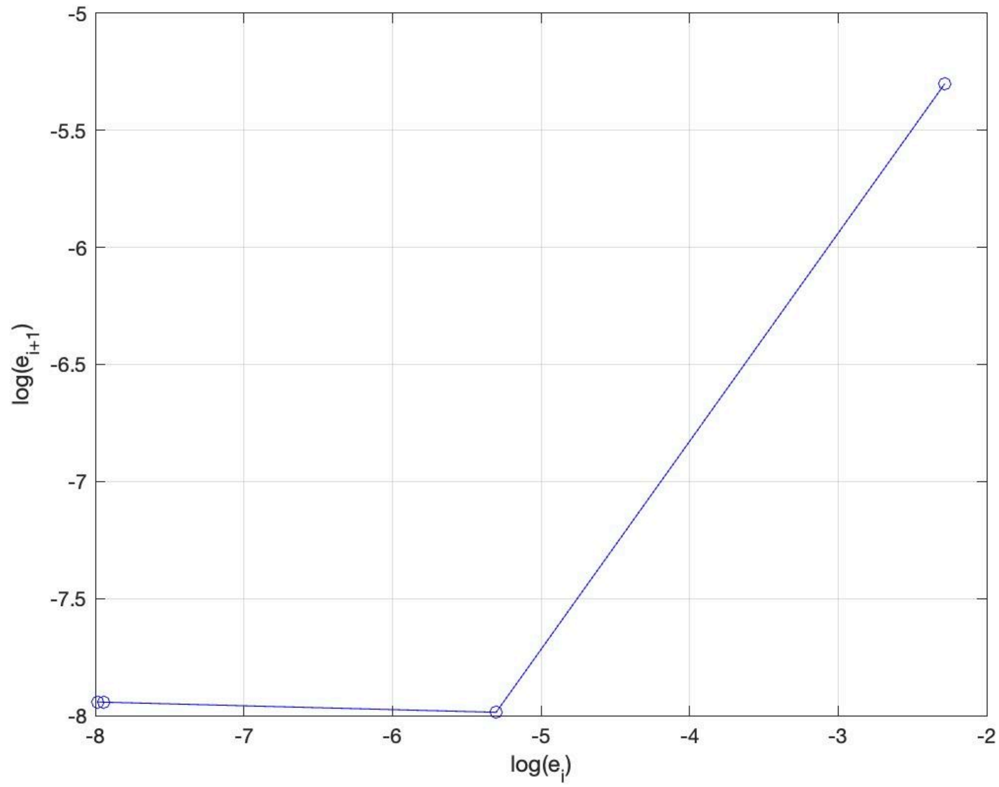


iv. $\boldsymbol{\alpha} = 1.62$. The numerical results partially match the theoretical prediction.(Further explanation is in the solution part). okay?

7. i. Initial bracketing interval is [0,1] and tolerance is 1e-6. Solution is 0.602355.

   ii. Errors = [1.020000000000000e-01, 4.983821699991786e-03, 3.401853949105416e-04, 3.554642387901819e-04, 3.554643642634803e-04]

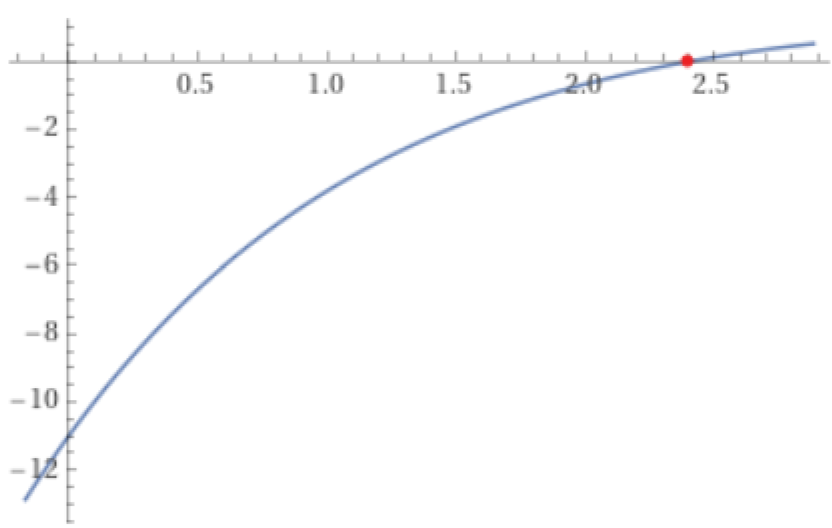
   iii. Slopes = [8.892550188875941e-01, -1.636600163791397e-02, 8.034438782010307e-06]



8. The distance downstream is 0.602355 kilometers where the oxygen level first falls to a reading of 5 mg/L.

Part B:

1. $f_2(x) = 4e^{-0.2x} - 15e^{-0.75x} = 0$

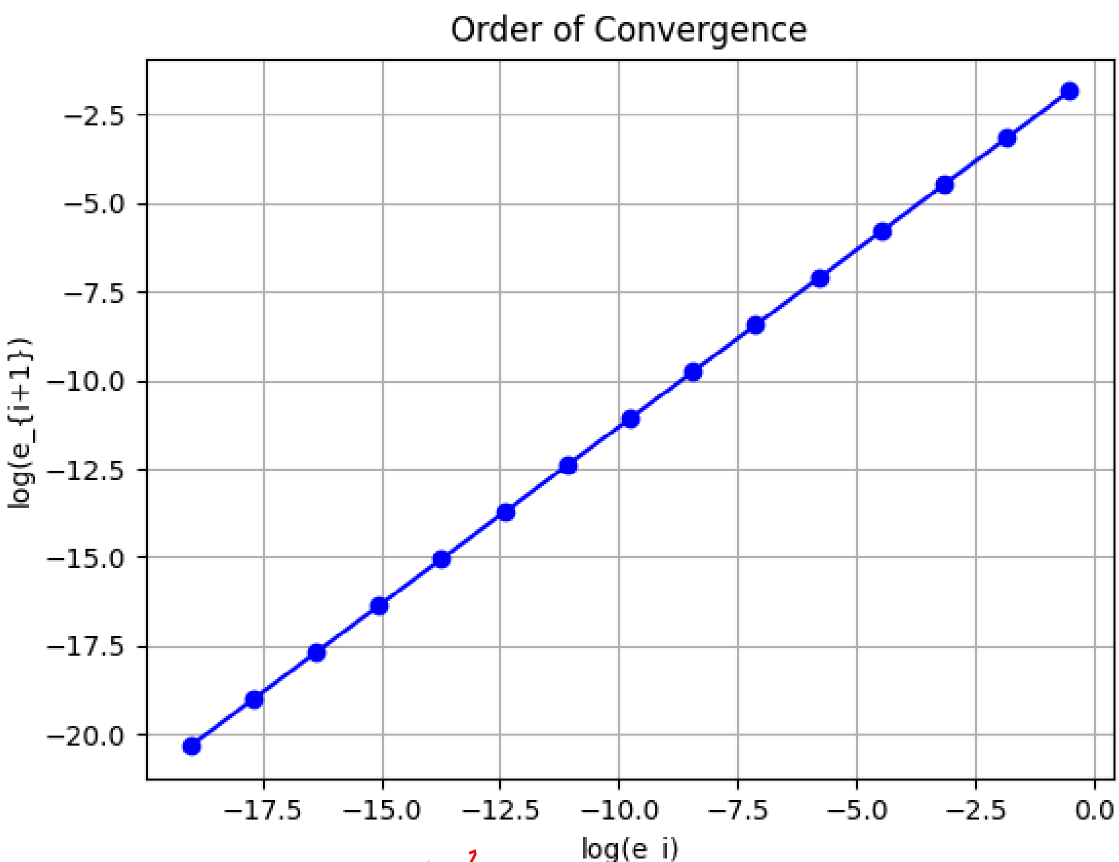
2. 

3. i. $g_2(x) = \frac{4}{15}x - \frac{4}{3}ln(\frac{4}{15})$ Why?

ii. Initial approximation is 3. Tolerance is 0.5e-6. The solution is 2.403192.

iii. Iteration errors = [5.96807564e-01, 1.59148684e-01, 4.24396490e-02, 1.13172397e-02, 3.01793060e-03, 8.04781493e-04, 2.14608399e-04, 5.72289076e-05, 1.52610431e-05, 4.06961259e-06, 1.08523112e-06, 2.89396057e-07, 7.71733744e-08, 2.05806590e-08, 5.48926815e-09, 1.46489754e-09]

iv. slopes = [0.9999999999857229, 0.9999999999464145, 0.9999999997991281, 0.9999999992468885, 0.9999999971757415, 0.9999999894065756, 0.9999999602755436, 0.9999998510512912, 0.9999994414468414, 0.9999979052553881, 0.9999921458059977, 0.9999705491014855, 0.9998895754163625, 0.9995859681914222]



why?

v. The theoretical slope is 1, and it matches my numerics.

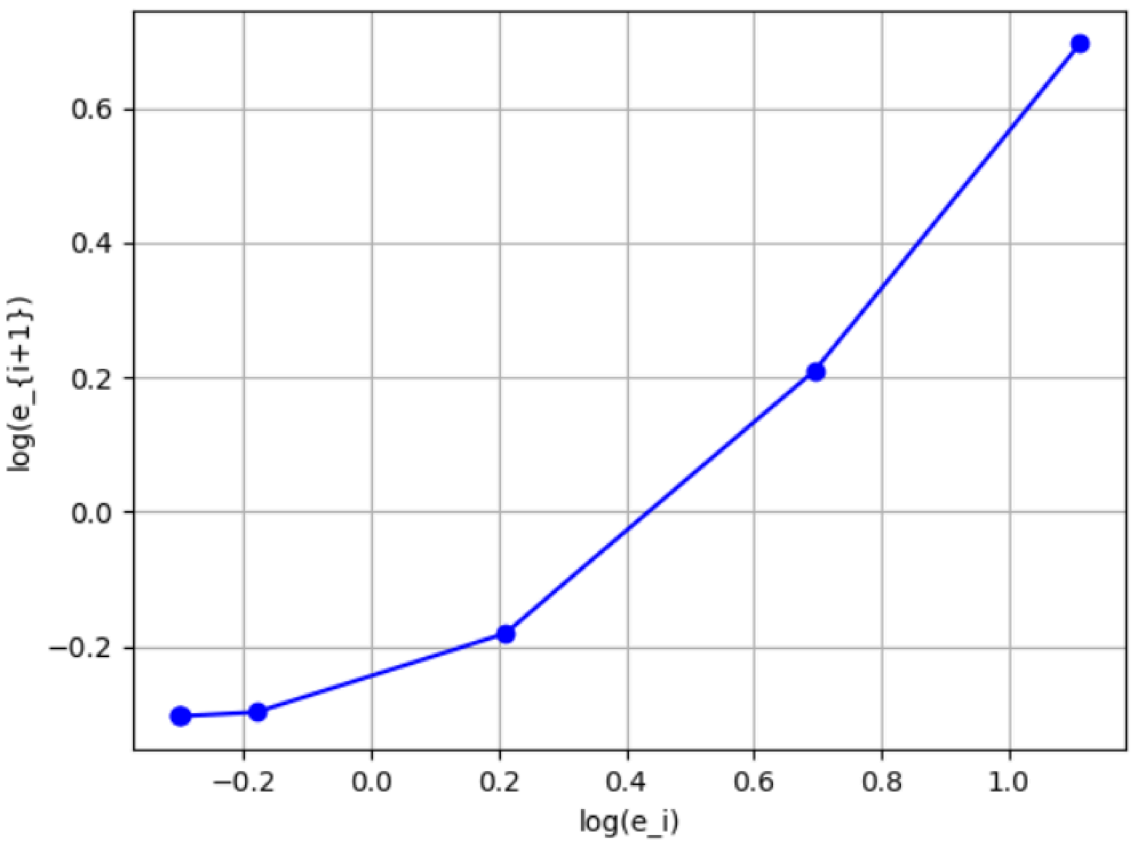
4. I. Initial approximation is 0.1. Tolerance is 0.5e-6. Root is 2.403192

Ii.

errors = [3.04159265, 2.00612141, 1.23422948, 0.83491739, 0.74265186 0.73840879, 0.73840022 ,0.73840022] your x-true is wrong

Iii.

slopes = [1.1671839016234098, 0.8046613801315908, 0.2996024385440079, 0.04892844718581346, 0.002025786147160902, 4.0712263401984136e-06]



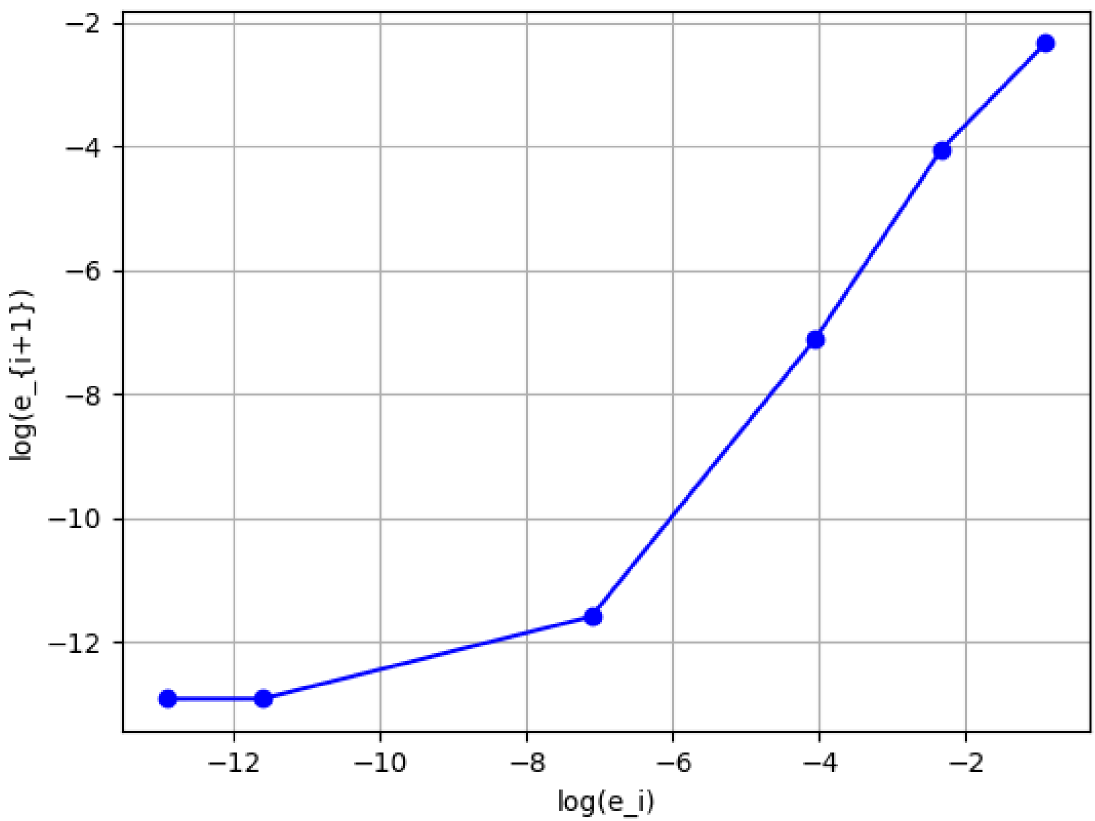
it's not   No, slopes ~ 0.74

Iv. Theoretical slope is 0. 396053. Numerical results partially match theory.

5. I. Initial approximations = 2 and 2.5. Tolerance = 1e-6. Solution r = 2.403192

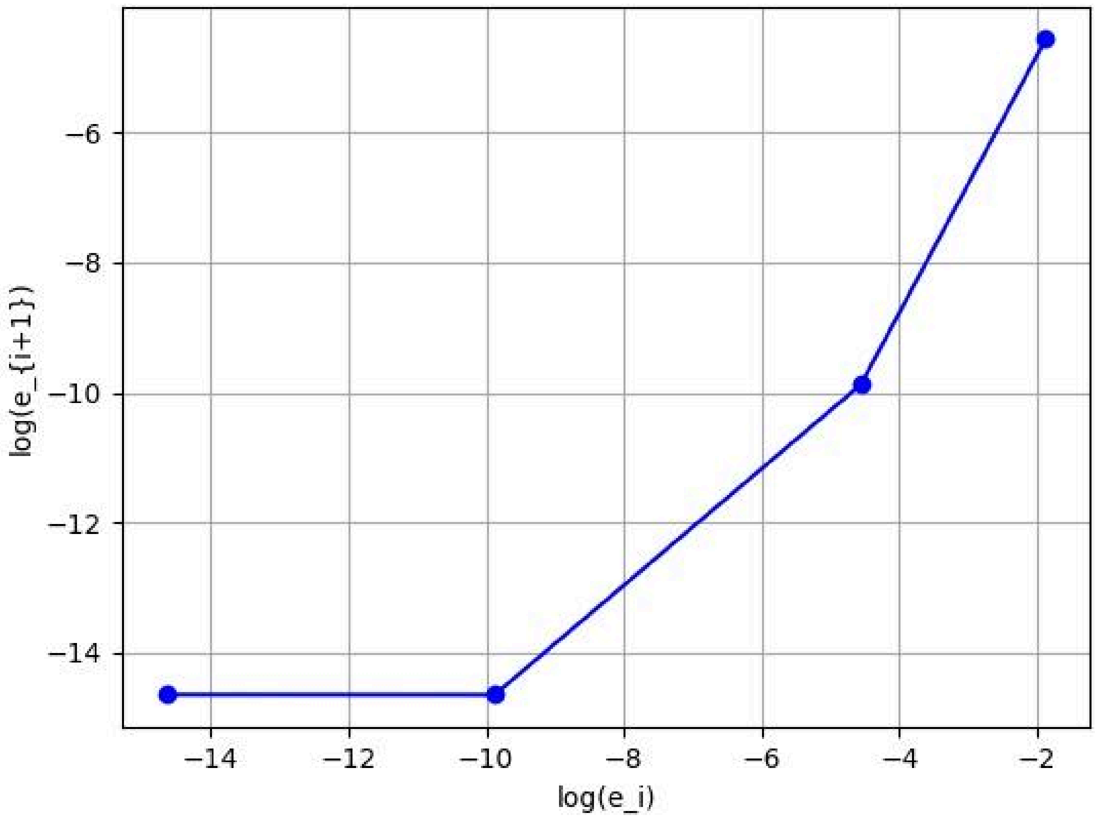
Ii.errors = [4.03190000e-01 9.68100000e-02 1.73388388e-02 8.14317326e-04 9.18560588e-06 2.43894948e-06 2.43633148e-06]

Iii.slopes = [1.205475838377804, 1.7783183291817457, 1.466381005333261, 0.2956868010123252, 0.0008099025383818838]



iv. $\alpha = 1.62$. The numerical results partially match the theoretical prediction.(Further explanation is in the solution part). Okay

6. i.Bracketing interval is [2,2.5] and tolerance is 5e-7. Solution is 2.403192.
   ii.Errors = [1.53192000e-01 1.04693976e-02 5.14029261e-05 4.35055046e-07 4.36331490e-071]
   iii.Slopes = [1.9813827594777593, 0.8975760095906502, -0.00061393572834582651]



7. i. Oxygen is at a minimum at 2.403192 kilometers downstream
   ii. 0.930303 mg/L
   iii. The fish will die painfully

Why painfully? Couldn't they just fall asleep?

	FPI	Newton	Secant	Newton Bisection	PartA(1)(2)(3) PartB(1)(2)	PartA(7) PartB(8)
Part A	Anjie Liu	Aalim Abdullah	Zun Cao	Anusha Singhal	Zun Cao Aalim Abdullah	Anusha Singhal
Part B	Anjie Liu	Aalim Abdullah	Zun Cao	Anusha Singhal	Zun Cao Aalim Abdullah	Anjie Liu Zun Cao

Solution & Discussion:

Part A: What is the distance downstream where the oxygen level first falls to a reading of 5 mg/L?

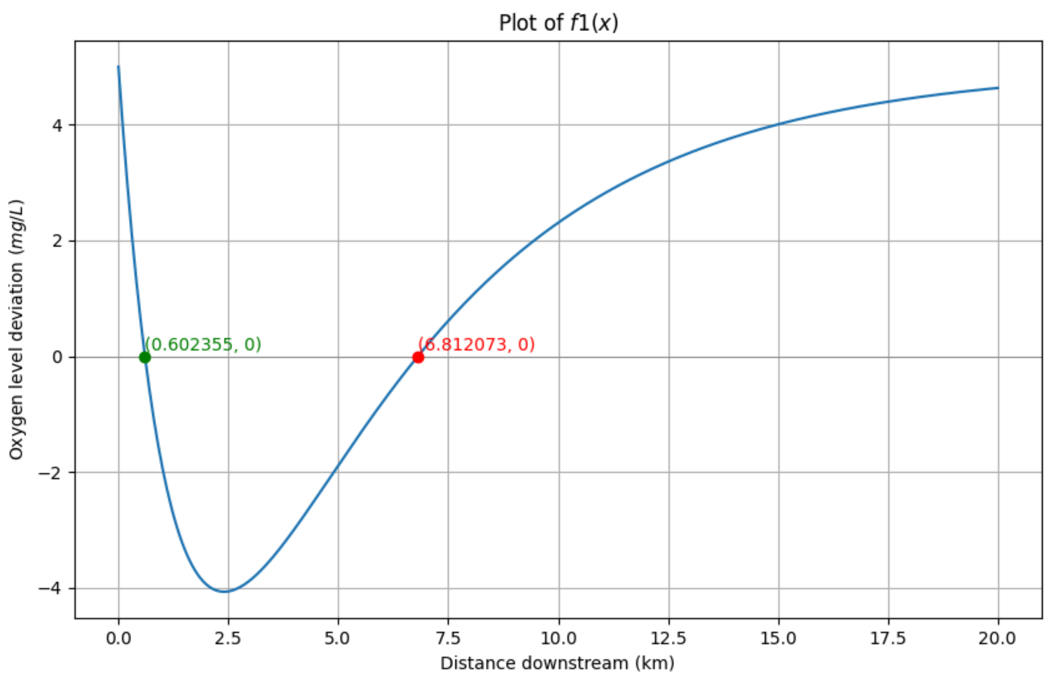
1. **What equation are you solving? Why?**

   We are solving the equation $c = 10 - 20(e^{-0.2x} - e^{-0.75x})$ where c is the oxygen level (mg/L) and x is the distance downstream in kilometers. In this case, c = 5 mg/L. That is, $5 = 10 - 20(e^{-0.2x} - e^{-0.75x})$. This can be rearranged to find the root x of the equation:

   $f_1(x) = 5 - 20(e^{-0.2x} - e^{-0.75x}) = 0$

2. **Plot the function f1 (corresponding to your equation).**

The plot of f1(plot by Python):



3. Which root r of f1 are you finding? Why?

The first root 0.602355 along the x-axis. This is where oxygen first falls to 5 mg/L.

4. For the Fixed Point Iteration (fixedPoint err):

i. **What is your choice of function g1 with a fixed point at r? Motivate your choice.**

$g_1(x) = -\frac{4}{3}ln(e^{-0.2x} - \frac{1}{4}) = x$

Okay next time put above

This function is chosen because $g_1'(x) = -\frac{16}{15e^{\frac{x}{5}}-60}$ and $|g_1'(0.5)| = 0.368 < 1$.

Other functions like $g(x) = -5ln(e^{-0.75x} + \frac{1}{4})$ do not work because

$g'(x) = \frac{16}{e^{\frac{3x}{4}}+4}$ and $|g'(0.5)| > 1$.

ii. **What are your initial approximation and tolerance? Please choose them such that the method takes at least 4-5 iterations. Find the solution with 6 correct decimal places.**

Initial approximation is 0.5. Tolerance is 0.5e-6. The solution is 0.602355.

13 total iterations are needed to reach the desired solution with my choice of initial assumption and tolerance.

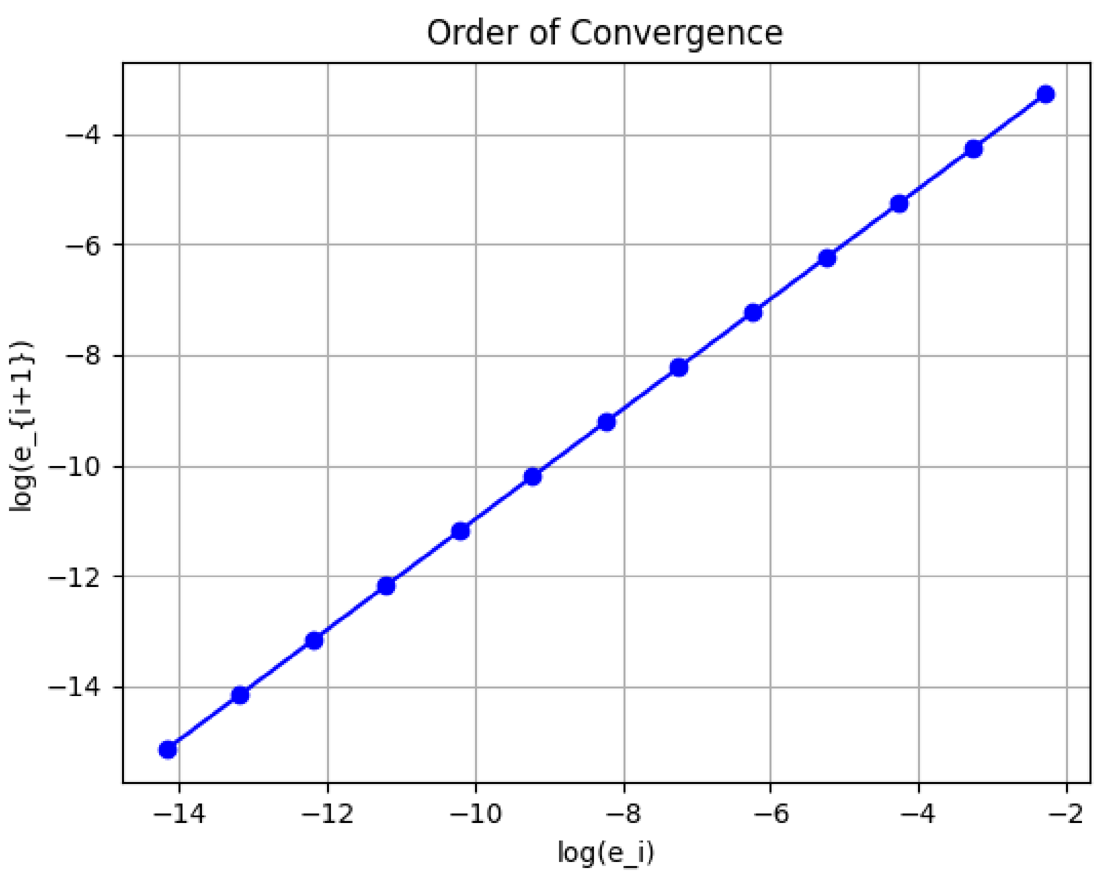
iii. What are the iteration errors?

Iteration errors = [1.02355464e-01, 3.78644098e-02, 1.40422361e-02, 5.21249458e-03, 1.93555399e-03, 7.18821110e-04, 2.66966707e-04, 9.91519119e-05, 3.68254399e-05, 1.36771578e-05, 5.07977138e-06, 1.88665557e-06, 7.00714544e-07, 2.60249355e-07]

iv. Include the graph and the vector with slopes described in the project description above.

slopes = [0.99748791386845, 0.9990615786462317, 0.9996507161185588, 0.9998701706089467, 0.9999517663637171, 0.9999820837851467, 0.999993345544567, 0.9999975284553999, 0.9999990820319946, 0.9999996590795894, 0.9999998732073845, 0.9999999530935259]

why diff 2



v. What is the slope that the theory predicts for FPI? Why? Does it match your numerics?

$\lim_{i\to\infty} \frac{ln(e_{i+1})}{ln(e_i)} = 1$. The slope that the theory predicts for FPI is 1 because

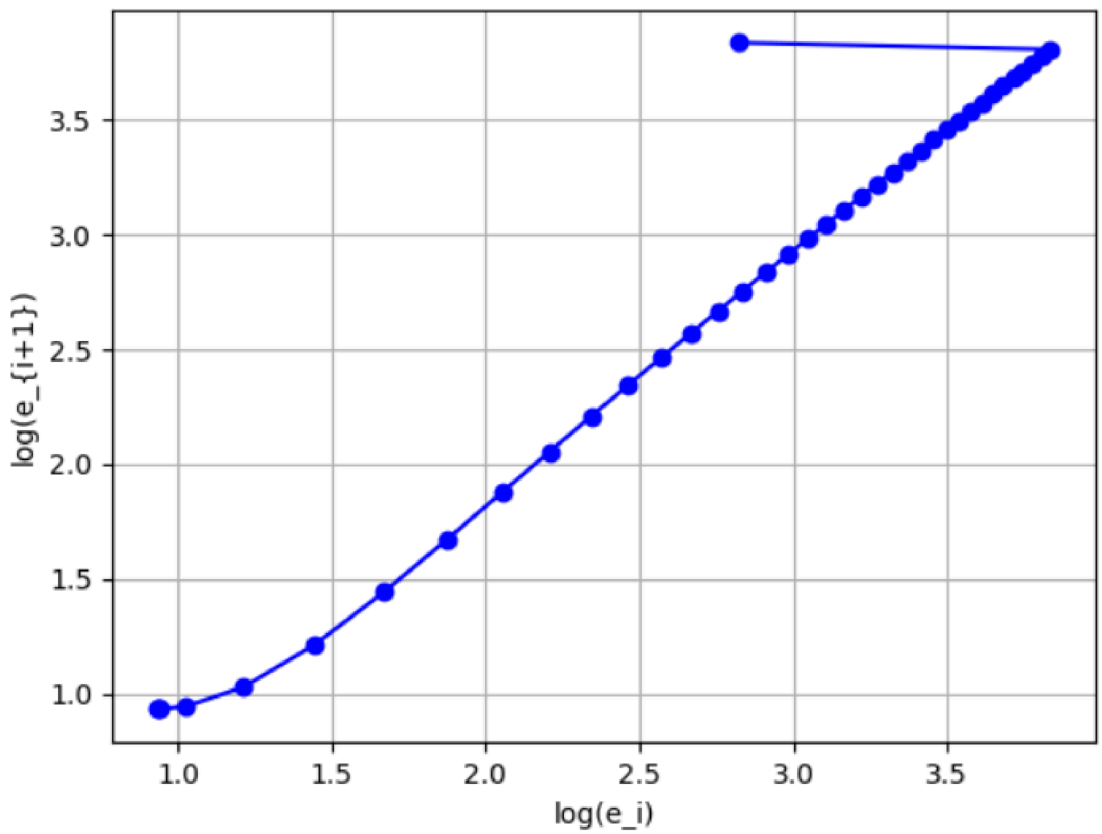
$e_{i+1} \approx Se_i \Rightarrow ln(e_{i+1}) \approx ln(Se_i) = ln(S) + \alpha\, ln(e_i)$, $where\ \alpha = 1$

OKay should be above

5. **For the Newton method (newton err):**
   i. What are your initial approximation and tolerance? Please choose them such that the method takes at least 4-5 iterations. Find the solution with 6 correct decimal places.
   The initial approximation is 20. Tolerance is 0.5e-6. Solution is r = 0.602355
   ii. What are the iteration errors?
   errors = [16.85840735 46.39332525 45.05999191 43.72665858 42.39332525 41.05999191
   39.72665858 38.39332525 37.05999192 35.72665859 34.39332528 33.05999198
   31.72665871 30.39332553 29.05999249 27.72665979 26.39332777 25.05999715
   23.72666948 22.39334792 21.06003905 19.72675654 18.39352872 17.06041421
   15.72753416 14.39513825 13.06373935 11.73438587 10.40920672 9.09248271
   7.79269735 6.5262924 5.32403641 4.24027606 3.36125044 2.79237835
   2.56958705 2.53972437 2.53923732 2.53923719]

   iii. Include the graph and the vector with slopes described in the project description above.
   slopes = [-0.02880634677808675, 1.03003899573557, 1.0309694261670084,
   1.0319593470632242, 1.0330146534278473, 1.0341420458322232,
   1.035349172871539, 1.0366448049248533, 1.0380390475100216,
   1.0395436050988307, 1.0411721097232423, 1.04294053340162, 1.0448677097687558,
   1.046975998854607, 1.0492921403297963, 1.0518483552105757,
   1.0546837738600612, 1.0578462871035232, 1.061394930493096,
   1.0654029006641879, 1.0699612221573163, 1.0751828284979705,
   1.0812061580243377, 1.0881957756290355, 1.0963338880903808,
   1.1057884708009187, 1.1166257081942492, 1.1285946560301106,
   1.1406232046795077, 1.1496627534753725, 1.1480659505532813,
   1.117854825929891, 1.0207024761094794, 0.7981366745073677,

0.4484347120856944, 0.14058758206364105, 0.016407089730401923, 0.0002618781806765]



iv. What is the slope that the theory predicts for Newton's method? Why? Does it match your numerics?

The theory predicts the slope to be $M = \lim_{i \to \infty} \frac{e_{i+1}}{e_i^2} = \left|\frac{f''(r)}{2f'(r)}\right| = 0.093110$ for r = 0.537484 ,$f'(x) = 4e^{-0.75x} - 15e^{-0.2x}$ *and* $f''(x) = 11.25e^{-0.75x} - 0.8e^{-0.2x}$..

No, 2.

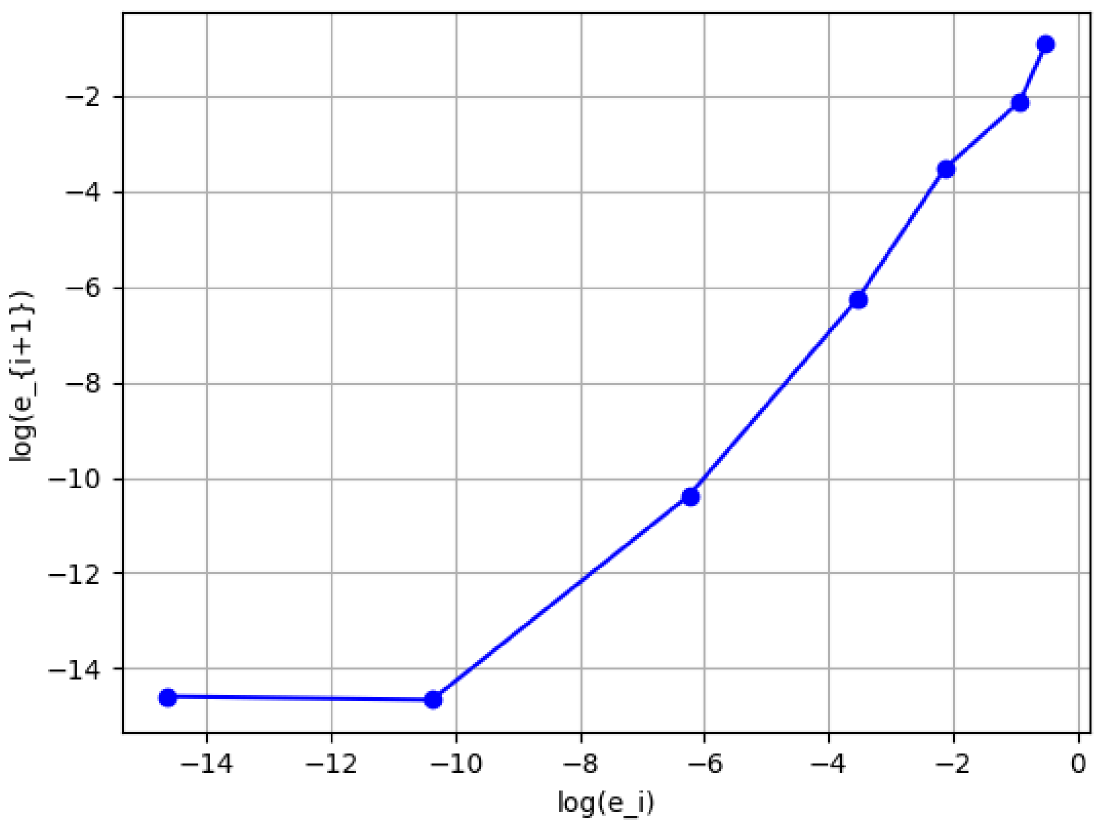
Since this is less than infinity and $f'(r) \neq 0$, Newtons method converges quadratically here. This does not exactly match my numerics as the slope is high around 1 and then stays there above 1 for most of the iterations. This could be due to the initial guess being far from the root in the very beginning of the method. Also the negative slope could highlight some numerical errors. However at around 0.14 and 0.01 the slope is close to the theoretical prediction which shows that the method does align somewhat with the theory. Overall, the numerics do not entirely match the theory but does show some little correlation.

Does not match what you have above. This is much better answer

6. **For the secant method (secant err):**
   i. What are your two initial approximations and tolerance? Please choose them such that the method takes at least 4-5 iterations. Find the solution with 6 correct decimal places.
   Two initial approximations are 0 and 1. Tolerance is 1e-6. (With iterations of 6). The solution is r = 0.602355.

   ii. What are the iteration errors?
   errors = [6.02355000e-01 3.97645000e-01 1.19428602e-01 2.95769135e-02
   1.94478389e-03 3.11466105e-05 4.32283129e-07 4.64364792e-07]

   iii. Include the graph and the vector with slopes described in the project description above.
   slopes = [2.8964065548583515, 1.1603567639246903, 1.9501291496019482, 1.5188974304279332, 1.0346328685390078, -0.016736803523302732]



   iv. What is the slope that the theory predicts for secant method? Why?

Does it match your numerics?

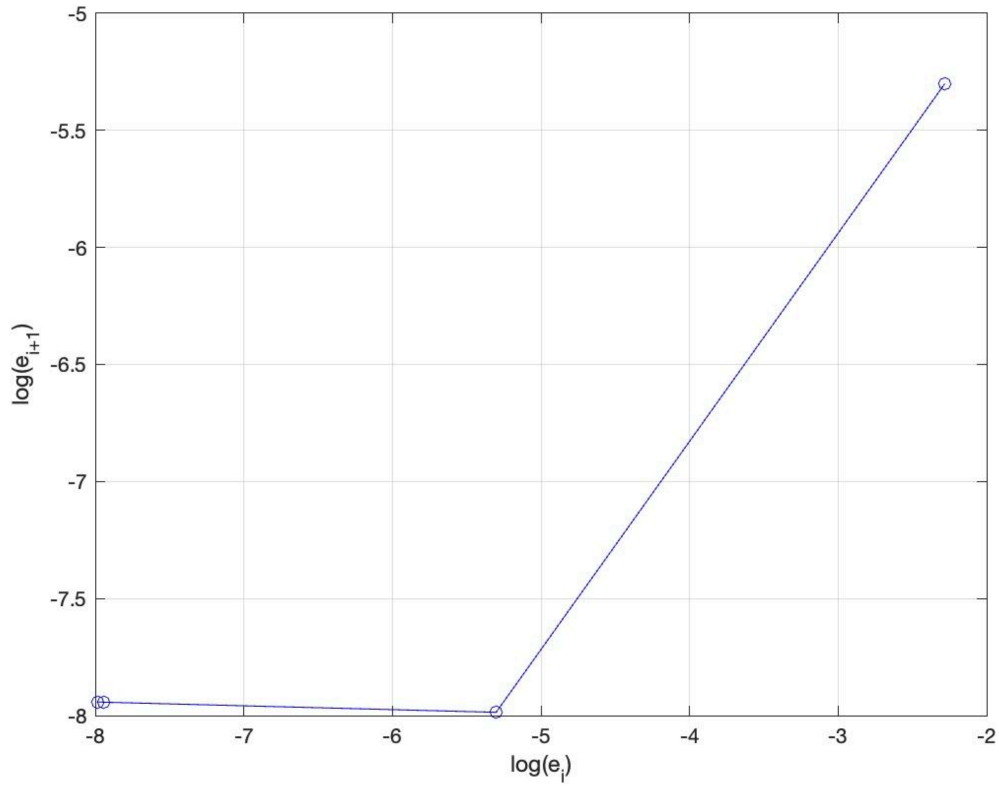
The theoretical convergence rate of the secant method is predicted by the concept of order of convergence, which for the secant method is $\boldsymbol{\alpha} = 1.62$. This is because the convergence rate of the secant method is determined by the limit of the ratio of successive errors as the number of iterations goes to infinity. According to the theorem,

$e_{i+1} \approx |\frac{f''(r)}{2f'(r)}|^{\alpha-1} e_i^{\alpha}$, where $\alpha = \frac{1+\sqrt{5}}{2} \approx 1.62$. This kind of convergence is superlinear.

From the picture and the number of slopes generated from Python code, while the average trend of the slopes might not clearly converge to $\boldsymbol{\alpha}$, individual slopes that are close to 1.62 would indicate moments where the method's convergence behavior aligns with theoretical expectations.

Specifically, the initial slopes are higher, with values such as 2.8964065548583515, which could result from the initial guesses being relatively far from the root, leading to larger initial corrections. Furthermore, the variation and especially the negative slope (-0.016736803523302732) in the final iterations suggest numerical instability or significant reduction in error, leading to less predictable slope calculations. This is common as errors approach machine precision and should not detract from the overall effectiveness of the method. Some of the values are indeed close to 1.62, such as 1.5188974304279332 and 1.0346328685390078. If the true value is more accurate, it will be more close to it. Therefore, the numerical results partially match the theoretical prediction.

7. **For the Newton-Bisection method (newtonBisection err):**
   i. What are your initial bracketing interval and tolerance? Please choose them such that the method takes at least 4-5 iterations. Find the solution with 6 correct decimal places.
   The initial bracketing interval is [0,1] and the tolerance is 1e-6. This takes 4 interactions and the solution is 0.602355.
   ii. What are the iteration errors?
   Errors = [1.020000000000000e-01, 4.983821699991786e-03, 3.401853949105416e-04, 3.554642387901819e-04, 3.554643642634803e-04]
   iii. Include the graph and the vector with slopes described in the project description above.
   Slopes = [8.892550188875941e-01, -1.636600163791397e-02, 8.034438782010307e-06]

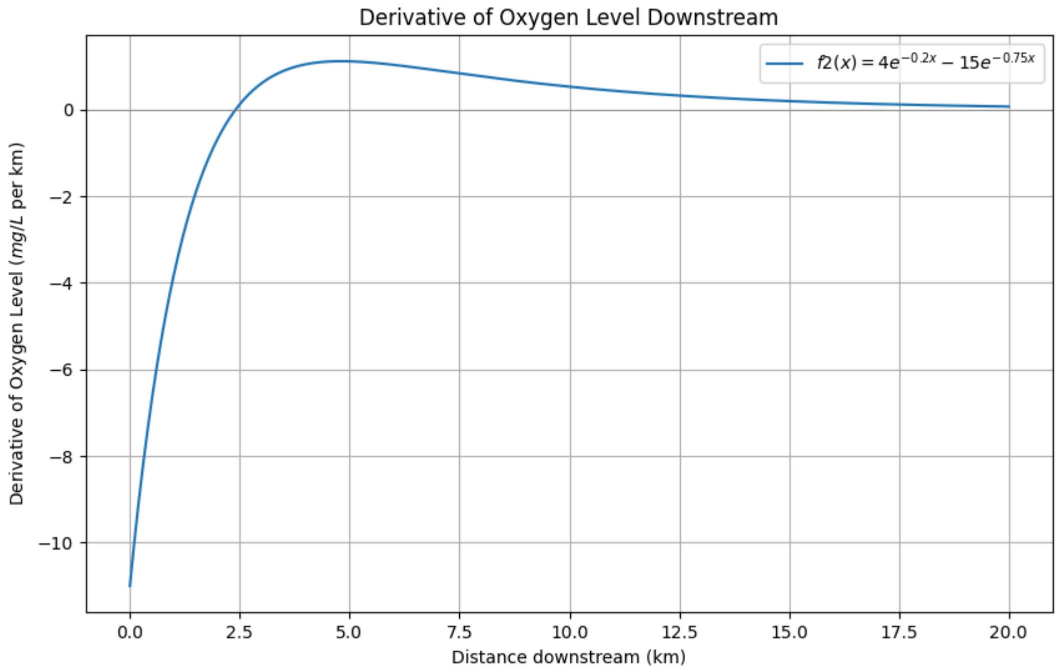


8. **What is the distance downstream where the oxygen level first falls to a reading of 5 mg/L?**
   The distance downstream is 0.602355 kilometers where the oxygen level first falls to a reading of 5 mg/L.

Part B: What is the distance downstream at which the oxygen is at a minimum? What is the concentration at that location? What will happen to the fish?

1. **What equation are you solving? Why?**
   Since in this question we aim to find the distance downstream where the oxygen level is at a minimum, we need to find where the derivative of the oxygen concentration equation with respect to distance, x, is zero. Given the equation for oxygen level c is:
   $c = 10 - 20(e^{-0.2x} - e^{-0.75x})$
   We should solve for f'(x) = 0 for the purpose of finding the minimum. The first derivative is:
   $f'(x) = 4e^{-0.2x} - 15e^{-0.75x} = 0$, which is the function f2 we need to solve.

2. **Plot the function f2 (corresponding to your equation).**
   The plot of f2(plot by Python):



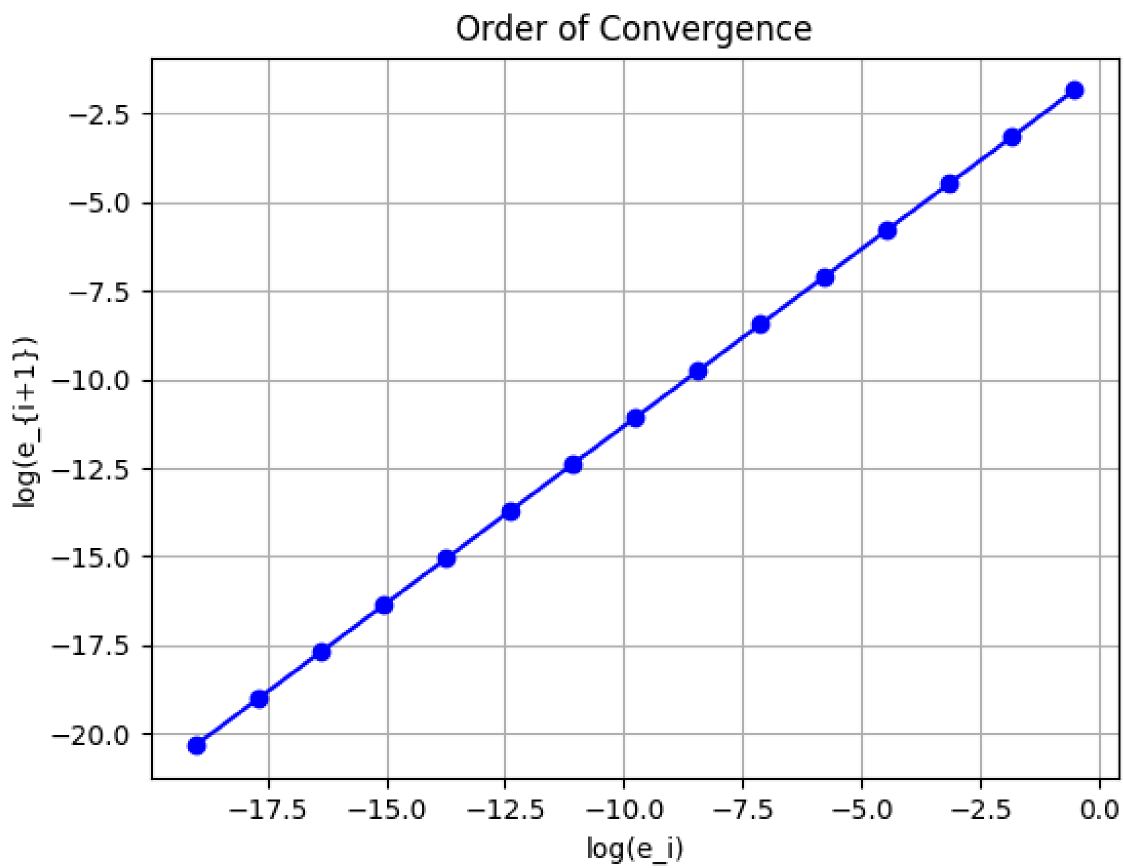
3. **For the Fixed Point Iteration (fixedPoint err):**
   i. What is your choice of function g2 with a fixed point at the root of f2? Motivate your choice.
   $g_2(x) = \frac{4}{15}x - \frac{4}{3}ln(\frac{4}{15}) = x$ is chosen because $|g_2'(x)| = \frac{4}{15} < 1$ which tells us that it might converge to the root linearly. Another possible $g(x)$ is $\frac{15}{4}x - 5ln(\frac{15}{4})$, but $|g'(x)| = \frac{15}{4} > 1$, which implies divergence.

   ii. What are your initial approximation and tolerance? Please choose them such that the method takes at least 4-5 iterations. Find the solution with 6 correct decimal places.
   Initial approximation is 3. Tolerance is 0.5e-6. The solution is 2.403192.
   13 total iterations are needed to reach the desired solution with my choice of initial assumption and tolerance.

   iii. What are the iteration errors?
   Iteration errors = [5.96807564e-01, 1.59148684e-01, 4.24396490e-02, 1.13172397e-02, 3.01793060e-03, 8.04781493e-04, 2.14608399e-04, 5.72289076e-05, 1.52610431e-05, 4.06961259e-06, 1.08523112e-06, 2.89396057e-07, 7.71733744e-08, 2.05806590e-08, 5.48926815e-09, 1.46489754e-09]
   The iteration errors are getting smaller linearly, which is as expected.

iv. Include the graph and the vector with slopes described in the project description above.

slopes = [0.9999999999857229, 0.9999999999464145, 0.9999999997991281, 0.9999999992468885, 0.9999999971757415, 0.9999999894065756, 0.9999999602755436, 0.9999998510512912, 0.9999994414468414, 0.9999979052553881, 0.9999921458059977, 0.9999705491014855, 0.9998895754163625, 0.9995859681914222]



v. What is the slope that the theory predicts for FPI? Why? Does it match your numerics?

$\lim_{i \to \infty} \frac{ln(e_{i+1})}{ln(e_i)} = 1$. The slope that the theory predicts for FPI is 1 because

$e_{i+1} \approx Se_i \Rightarrow ln(e_{i+1}) \approx ln(Se_i) = ln(S) + \alpha\, ln(e_i),\ where\ \alpha = 1$. This matches my numerics.

4. **For the Newton method (newton err):**

i. What are your initial approximation and tolerance? Please choose them such that the method takes at least 4-5 iterations. Find the solution with 6 correct decimal places.

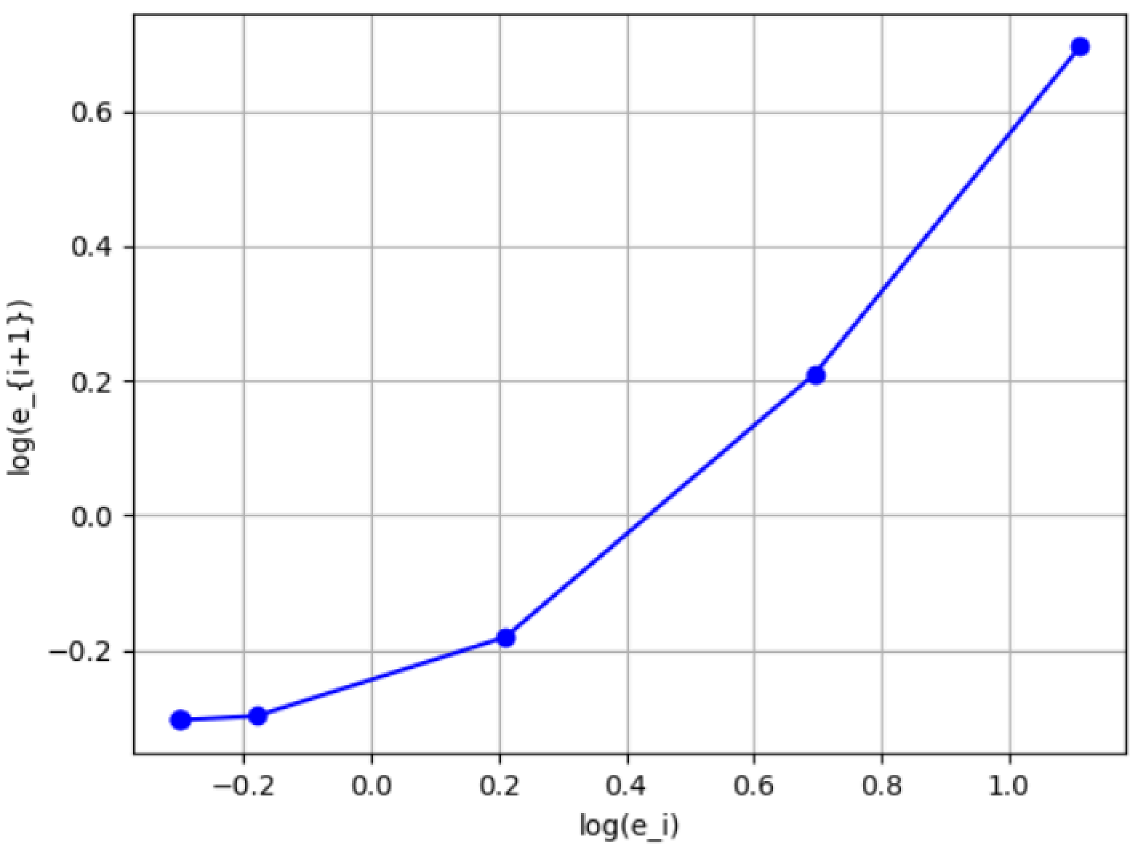
Initial approximation is 0.1. Tolerance is 0.5e-6. Root is 2.403192

ii. What are the iteration errors?

errors = [3.04159265, 2.00612141, 1.23422948, 0.83491739, 0.74265186 0.73840879, 0.73840022 ,0.73840022]

iii. Include the graph and the vector with slopes described in the project description above.

slopes = [1.1671839016234098, 0.8046613801315908, 0.2996024385440079, 0.04892844718581346, 0.002025786147160902, 4.0712263401984136e-06]



iv. What is the slope that the theory predicts for Newton's method? Why? Does it match your numerics?

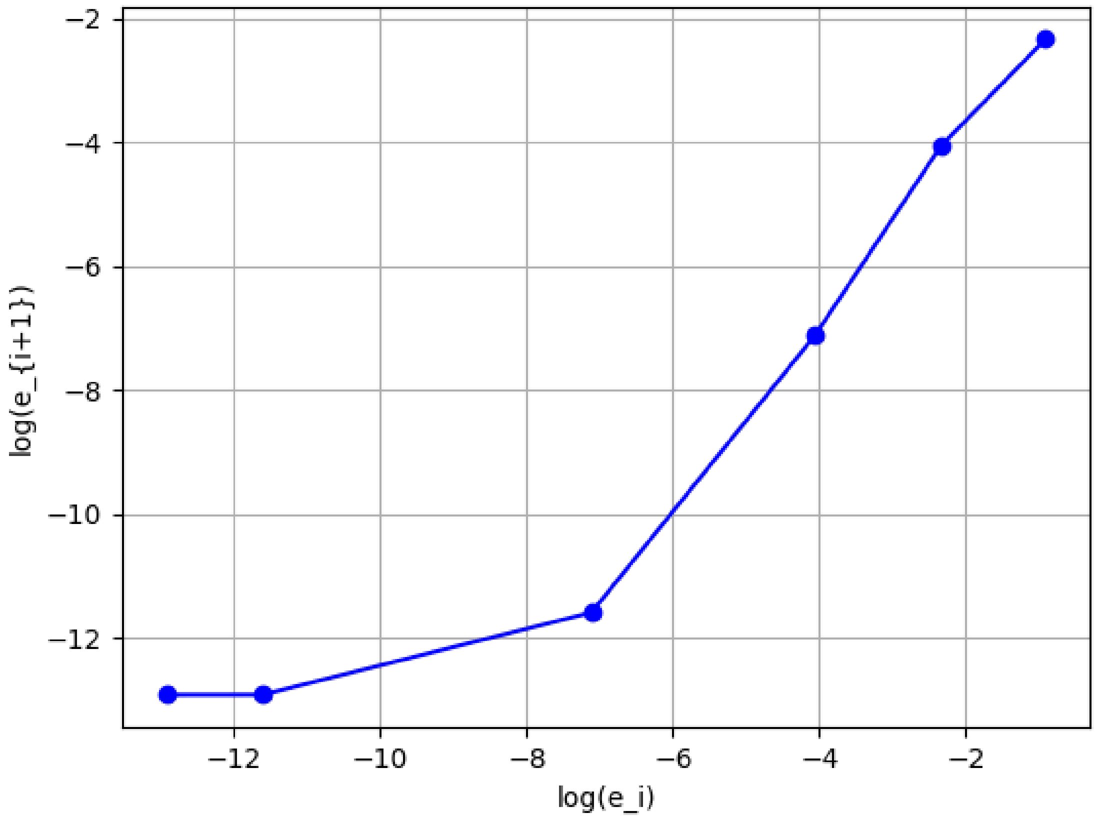
The theory predicts the slope to be $M = \lim_{i \to \infty} \frac{e_{i+1}}{e_i^2} = \left|\frac{f''(r)}{2f'(r)}\right| = 0.396053$ for r = 2.403192,

$f'(x) = 8.4375e^{-0.75x} + 0.16e^{-0.2x}, and\ f''(x) = -0.032e^{-0.2x} + 6.32813e^{-0.75x}$.
Since M less than infinity and $f'(r) \neq 0$, Newton's method converges quadratically here. While the slope is quite high in the beginning at greater than 1, the slopes quickly start to decrease quadratically towards zero. At around 0.80 and 0.299 the slope is close to the theoretical prediction which shows that the method does align somewhat with the theory. The numerics here somewhat match the theory.

5. **For the secant method (secant err):**
i. What are your two initial approximations and tolerance? Please choose them such that the method takes at least 4-5 iterations. Find the solution with 6 correct decimal places.
Two initial guesses are 2 and 2.5. Tolerance is 1e-6(With iterations of 5).
The solution is r = 2.403192.

ii.What are the iteration errors?
errors = [4.03190000e-01 9.68100000e-02 1.73388388e-02 8.14317326e-04
9.18560588e-06 2.43894948e-06 2.43633148e-06]

iii. Include the graph and the vector with slopes described in the project description above.
slopes = [1.205475838377804, 1.7783183291817457, 1.466381005333261, 0.2956868010123252, 0.0008099025383818838]



iv. What is the slope that the theory predicts for secant method? Why? Does it match your numerics?

Same as part a, the theory predicts that the convergence rate of the secant method is $\alpha = 1.62$. This is due to the secant method's reliance on linear interpolation between two points to approximate the derivative, which leads to a superlinear convergence rate. This ratio emerges from the formula for the secant method's convergence because it optimally balances the reduction of error in each step given the method's reliance on the two most recent approximations.(quite same as answer 6.iv. of part a)

From the Python plot and the slopes, numerical results show variability around 1.62. Specifically, some of the slopes, 1.778318 and 1.466381, are relatively close to α; The lower values (0.295687, 0.000810) towards the end suggest that as we get closer to the root, the differences in successive errors become smaller, which can lead to a decrease in the calculated slope. Thus, the numerical results partially match the theoretical prediction.

6. **For the Newton-Bisection method (newtonBisection err):**
   i. What are your initial bracketing interval and tolerance? Please choose them such that the method takes at least 4-5 iterations. Find the solution with 6 correct decimal places.

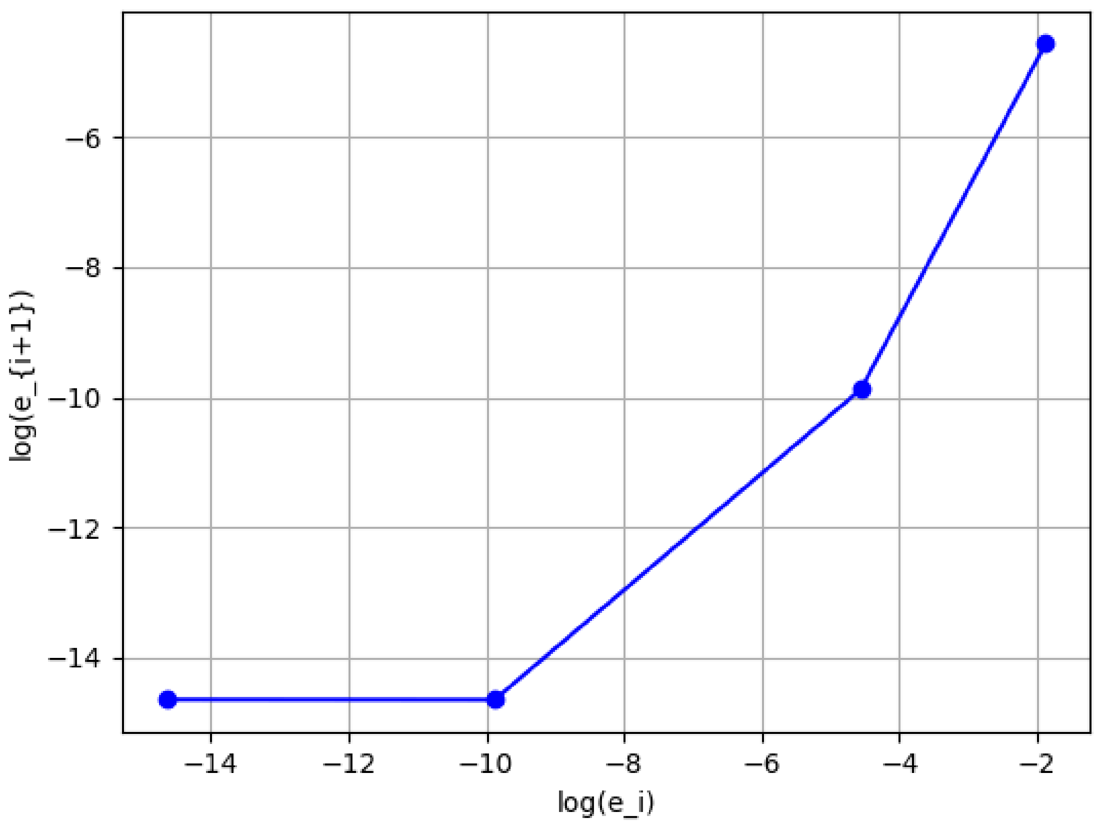
The bracketing interval is [2,2.5] with a tolerance of 5e-7. This also required 4 iterations to get a solution of 2.403192.

ii. What are the iteration errors?

Errors = [1.53192000e-01, 1.0,04693976e-02, 5.14029261e-05, 4.35055046e-07, 4.36331490e-07]

iii. Include the graph and the vector with slopes described in the project description above.

Slopes = [1.9813827594777593, 0.8975760095906502, -0.00061393572834582651]



7. **Answer the following questions.**

i. What is the distance downstream at which the oxygen is at a minimum?

Oxygen is at a minimum at 2.403192 kilometers downstream.

ii. What is the concentration at that location?

To find the corresponding oxygen concentration at the calculated downstream distance x=2.403192, we plug it in into $c = 10 - 20(e^{-0.2x} - e^{-0.75x})$, which equals 0.930303 mg/L.

iii. What will happen to the fish at the found concentration?

We concluded that fish will die painfully based on the given information that levels of oxygen below 5 mg/L are generally harmful to game-fish such as trout and salmon. Since the minimum oxygen concentration of 0.930303 mg/L is way below the 5 mg/L threshold, we determined that fish could hardly survive under this circumstance.

Part C:

1. **Did you struggle with any part of the group project?**
   Yes, we encountered a few challenges during the project. One struggle was aligning everyone's schedules for group meetings, given our differing class schedules and personal commitments. This sometimes delayed decision-making and progress checks.
   In our group project, we encountered a challenge with the Newton-Bisection method when Anusha found that her initial guess interval was causing the code to fail. Working with Zun, initially the issue was thought to be the bracketing interval, but when Anusha further inspected her code, it turned out to be an arithmetic error due to misplaced parentheses. Upon fixing this, the original interval generated the accurate solution and code.
2. **What do you think we should change so our next sessions run better?**
   I suggest that the deadlines for homework and group projects aren’t together. :)
3. **Do you have any conceptual questions that didn’t get answered?**
   No!
4. **Feel free to add any comments that you want to make but I haven’t listed.**
   The scenario is quite interesting. Overall, I believe our group worked well together, demonstrating a good blend of skills and a cooperative spirit. However, we could improve our project experience by starting our process earlier, allowing more time for analysis and addressing any unexpected challenges. Additionally, I think it would be beneficial for us to allocate more time for a retrospective discussion after project submission, to identify lessons learned and best practices that we can carry into future group projects.