Mathematics Year 1, Calculus and Applications I

Portfolio Marks Assessment 1

Georgi Angelov

For the purpose of completing the whole exercise, the Python programming language and its mathplotlib library were used.

GitHub repository: https://github.com/Georgi-G-Angelov/Numerical-Derivative-Approximation

1 Approximating derivatives

This exercise uses the 3 following methods of approximating a function's derivative and a point x_0:

$$(i) f`(1) \approx \frac{f(1+h)-f(1)}{h} := D_+f, h > 0.$$

$$(ii) f`(1) \approx \frac{f(1+h)-f(1)}{h} := Df, h < 0.$$

$$(iii) f`(1) \approx \frac{f(1+h)-f(1-h)}{2h} := Df, h > 0.$$

The objective is to evaluate how accurate the three formulas are in comparison to the actual value of the particular function. In order to do this, a known function is picked - $f(x) = x^3$. We know $f`(x) = 3x^2$ and therefore $f`(1) = 3$.

1. First, D_+f, D_-f, Df were calculated for $h = 1/2^n$ for $n = 1, 2, . . . , 10.$ Afterwards, the errors $\varepsilon_1 := |D_+f - f`(1)|, \varepsilon_2 := |D_-f - f`(1)|$ and $\varepsilon_3 := |Df - f`(1)|$ for each method of approximation was calculated.

These are the values for the three approximations:

h	D_+f	D_-f	Df
1	4.75	1.75	3.25
2	3.8125	2.3125	3.0625
3	3.390625	2.640625	3.015625
4	3.19140625	2.81640625	3.00390625
5	3.0947265625	2.9072265625	3.0009765625
6	3.04711914062	2.95336914062	3.00024414062
7	3.02349853516	2.97662353516	3.00006103516
8	3.01173400879	2.98829650879	3.00001525879
9	3.0058631897	2.9941444397	3.0000038147
10	3.00293064117	2.99707126617	3.00000095367

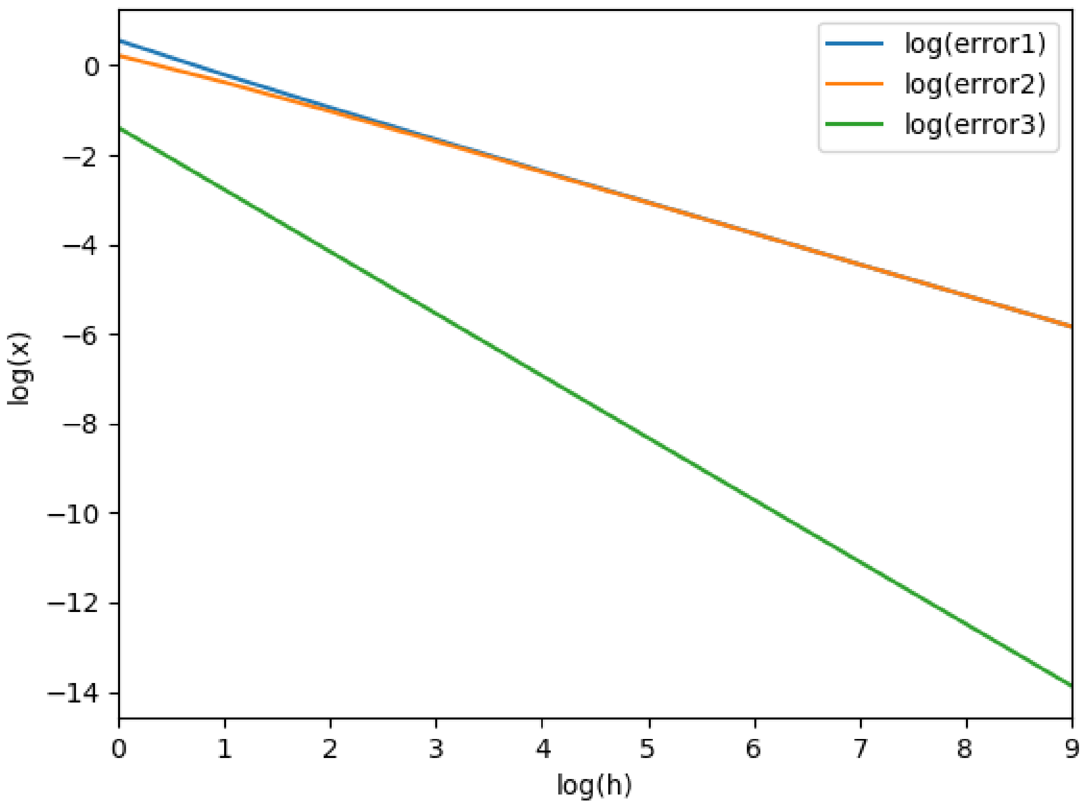
These are the error values in the three different cases:

h	ε_1	ε_2	ε_3
1	1.75	1.25	0.25
2	0.8125	0.6875	0.0625

3	0.390625	0.359375	0.015625
4	0.19140625	0.18359375	0.00390625
5	0.0947265625	0.0927734375	0.0009765625
6	0.047119140625	0.046630859375	0.000244140625
7	0.0234985351562	0.0233764648438	0.00006103515625
8	0.0117340087891	0.0117034912109	0.0000152587890625
9	0.00586318969727	0.00585556030273	0.00000381469726562
10	0.00293064117432	0.00292873382568	0.000000953674316406

The much higher accuracy of the third approximation method is already evident.

2. A log-log plot of $\log(\varepsilon_{1,2,3})$ versus $\log(\mathrm{h})$ was produced to visualize the data.



3. To conclude, by inspecting the visualized data provided on the plot, approximation scheme (iii) is much more accurate than (i) and (ii).

The error in schemes (i) and (ii) is in fact almost the same. It is evident from the graph that their rate of change is very close. It appears as if the slopes of the curves representing $\log(\varepsilon_1)$ and $\log(\varepsilon_2)$ are equal.

The error ε_3 in scheme (iii), however, drops much more quickly. The curve which represents $\log(\varepsilon_3)$ has a much steeper slope than the curves that represent $\log(\varepsilon_1)$ and $\log(\varepsilon_2)$. Therefore, we can conclude that scheme (iii) is in fact the most recommendable for approximating derivatives out of the three.

2 Solving a differential equation numerically

In this part, the following differential equation is considered:

$\frac{dy}{dx} = y, 0 < x \leq 1, y(0) = 1.$

The solution is well known, $\mathrm{y{=}exp(x)}$. We know that $\frac{dy}{dx} exp(x) = exp(x)$ and also the antiderivative of $\mathrm{exp(x)}$ is $\mathrm{exp(x)\ +\ c}$, where c is a constant. However, it is given that $\mathrm{y(0)\ =\ 1}$ and therefore $\mathrm{exp(0)\ +\ c\ =\ 1}$ which means that $\mathrm{c{=}0}$.

We define a discretisation of the interval $[0, 1]$ as follows. For an integer $N > 0$ that measures the number of grid points $\mathrm{x}_k = kh, h = \frac{1}{N}, k = 0, 1, 2, ..., N, x_0 = 0.$

The objective is that the value of $\mathrm{y(1)}$ is approximated by discrete values y_k at the grid points $\mathrm{k{=}1{,}2{,}..N}$.

1. Let’s look at scheme $(ii) f‘(1) \approx \frac{f(1 + h) f(1)}{h} := D_{-}f, h < 0$ from the previous part.

Therefore, after applying it to this problem, we get that

$$\frac{dy}{dx}(x_k) \approx \frac{y(x_{k+1}) - y(x_k)}{h}$$

However, since $\frac{dy}{dx} = y$, then $\frac{dy}{dx}(x_k) = y(x_k) = y_k.$

Therefore,

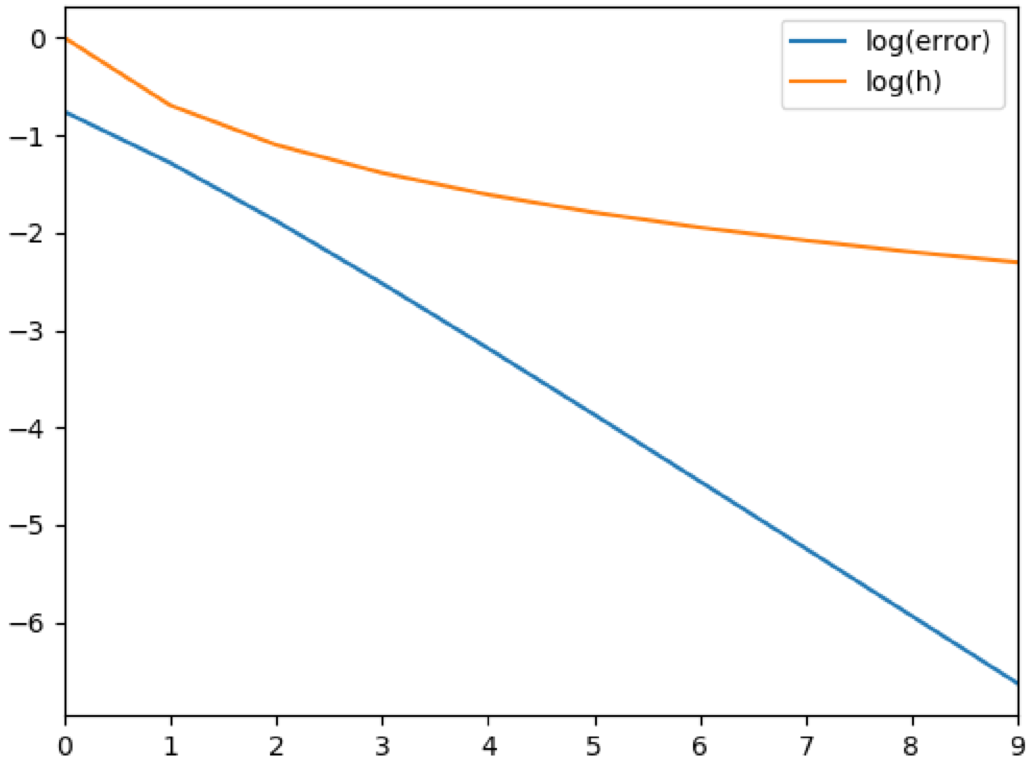
$$y_k = \frac{y_{k+1} - y_k}{h}$$
$$y_k h = y_{k+1} - y_k$$
$$y_k(1 + h) = y_{k+1}$$

and therefore,

$y_{k-1}(1 + h) = y_k$ or

$y_k = (1 + h) y_{k-1}.$

2. Using this information, we can calculate the wanted discrete values and plot the error $|y_N - y(1)|$ for N $= 2^n, n = 1, ..., 10.$



From this plot it is evident that the rate of drop of the error is much greater that the rate of decrease of h. As N get bigger and bigger, the error $|y_N - y(1)|$ decreases at an even greater rate. This can be deduced by looking at the slopes of the curves which represent $\log(\mathrm{h})$ and $log(|y_N - y(1)|)$ respectively.

3. We know that $y_k = (1+h)y_{k-1}$. We can now prove by induction that $y_k = (1+h)^k y_0$.
- $y_1 = (1+h)y_0$ is given.
- Let's assume that for some n, $y_n = (1+h)^n y_0$.
- We now have to show that $y_{n+1} = (1+h)^{n+1} y_0$.

But we know that $y_{n+1} = (1+h)y_n$ which is

$$(1+h)(1+h)^n y_0 = (1+h)^{n+1} y_0.$$

With this our induction is finished.

Therefore, we have that $y_k = (1+h)^k y_0 = (1+\frac{1}{N})^k y_0 = (1+\frac{1}{N})^k$.

Therefore, if we set N $\to \infty$, then

$$\lim_{N\to\infty} y_N = \lim_{N\to\infty} (1+\frac{1}{N})^N = y(1) = e.$$