14-01-2024

Numerical Analysis

Calculating integrals

Haka Kevin

Introduction

Integrals appear in almost every field of natural science and engineering. Most theoretical models use integrals in their calculations. Some uses of integrals are to calculate the center of mass of an object, to calculate the volume of a solid, to calculate the position, velocity, and acceleration of a body like planets, and much more. The challenge with integrals is that only a few problems are analytically solvable, and that's why we use numerical analysis to approximate complex integrals for which we have no solution. However, the ability of numerical analysis methods to approximate solutions for unsolvable problems comes with some costs, such as memory (which is not a big deal today), the time that those methods need to work, and the precision of the answer. Most methods are recursion, which means that theoretically they will approximate the true value after an infinite number of iterations. However, neither Simpson's 1/3 rule nor the Monte Carlo method which we are going to use in this assignment are recursive methods. Because of this, we cannot use the **Scarborough criterion** (1) to achieve the desired precision. We will instead conduct a graphical analysis to see how the method approximates the true value.

$$\varepsilon_s = \frac{1}{2} 10^{-n} \tag{1}$$

$$\int_0^1 \int_x^{2x} (x^2 + y^3)\, dy\, dx \tag{2}$$

In this assignment, we are going to solve the above double integral (2) in three different ways. First, we will solve it analytically. Next, we will solve it using the Simpson's 1/3 method, which is a numerical method. Finally, we will solve it using the Monte Carlo method, which is a simulation technique.

Analytical Calculation

$$\int_0^1 \int_x^{2x} (x^2 + y^3)\, dy\, dx = \int_0^1 \left[y \cdot x^2 + \frac{1}{4} y^4 \right]_x^{2x} dx =$$

$$\int_0^1 \left(2x^3 + 4x^4 - x^3 - \frac{1}{4} x^4 \right) dx = \int_0^1 \left(x^3 + \frac{15}{4} x^4 \right) dx =$$

$$\left[\frac{1}{4} x^4 + \frac{3}{4} x^5 \right]_0^1 = \frac{1}{4} + \frac{3}{4} = 1$$

$$\int_0^1 \int_x^{2x} (x^2 + y^3)\, dy\, dx = 1 \tag{3}$$

In our case, the double integral has an analytical solution and is easy to solve (equation 3). However, as we mentioned before, this is very rare for real problems. The true value that we found above will help us validate the results from the next two methods and then compare them. One problem that we will encounter with the next two methods is that the first integration has limits that are functions of x, but these methods work only with fixed limits.

Let’s start with our problem in this double integral, which is the non-fixed limits of the first integral. First of all, let’s generalize the problem to find a systematic method to approach it. We will set the limits of the first integral as functions of x, and the limits of the last integral will be constant. The truth is that we can generalize it much more with n successive integrals, where the limits of each integral can be functions of all yet unintegrated variables (4), with the last one having only constant limits. But for simplicity, we will continue with the double integral (5), because the geometric explanation that we will show next is impossible to understand in more than three dimensions.

$$\int_l^u \int_{g_1(x_1)}^{g_2(x_1)} \cdots \int_{g_{2(n-2)-1}(x_1,x_2,\ldots,x_{n-2})}^{g_{2(n-2)}(x_1,x_2,\ldots,x_{n-2})} \int_{g_{2(n-1)-1}(x_1,x_2,\ldots,x_{n-1})}^{g_{2(n-1)}(x_1,x_2,\ldots,x_{n-1})} f(x_1, x_2, \ldots, x_n)\, dx_n\, dx_{n-1} \cdots dx_1 \tag{4}$$

$$\int_{low}^{up} \int_{g(x)}^{h(x)} f(x, y)\, dy\, dx \tag{5}$$

It is not the purpose of this assignment to delve into what the integrals express, and that is why we will be satisfied with the area of the integration. In two dimensions (x, y), if the limits of both integrals are constant values, the area of the integration will be a rectangle (Figure 1). However, this is just the simplest case. As we can see in Figure 2, the limits of the y integral can be a function of x, which can give us a non-common integration area. In our case, where the integration limits for the y variable are h(x) = x and g(x) = 2x, the integration area for our double integration can be shown in Figure 3, which is the area between the two lines h(x) and g(x) (purple color).

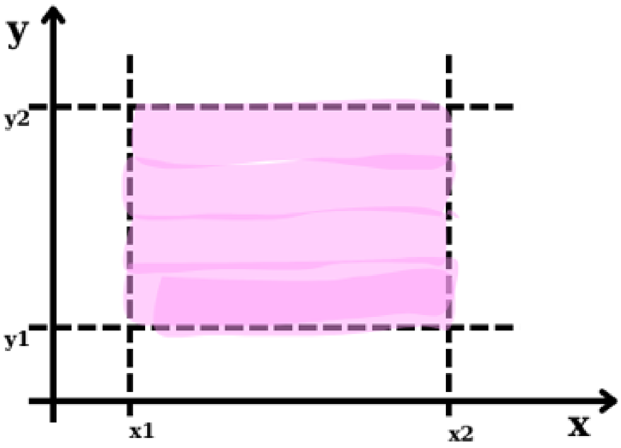

Figure 1 In this figure, we can see the area (rectangle) of integration in pink color for a double integration with constant limits: (x1, x2) for the x variable and (y1, y2) for the y variable.

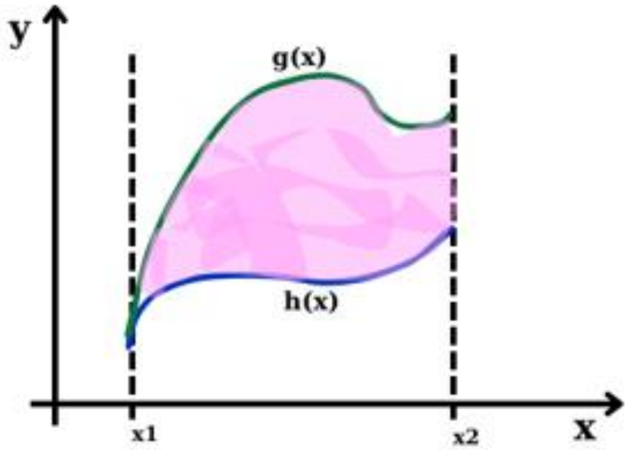

Figure 2 In this figure, we can see the area of integration in purple color for a double integration with limits: (x1, x2) for the x variable and (g(x), h(x)) for the y variable.

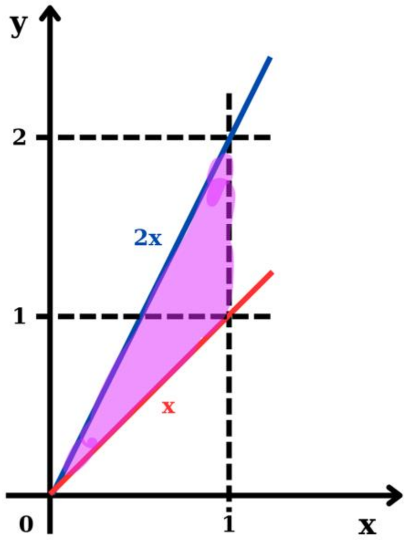

Figure 3 In this figure, we can see the area of integration in pink color for a double integration with limits: (0, 1) for the x variable and (x, 2x) for the y variable.

Numerical Calculation

Since we understood what happens with the limits of the y variable, it's time to start constructing the algorithm for numerical calculation. There are many methods that we can choose from, such as the trapezium rule, Simpson's rule 1/3, Simpson's rule 3/8, Boole's rule, and more. We will use Simpson's 1/3, which is better than the trapezium rule by two orders of magnitude and has the same accuracy as Simpson's 3/8. We could pick Boole's rule, which is two orders of magnitude better than Simpson's 1/3, but it's too complicated to implement. Moreover, our function is an increasing function as for x and y in the integration area (Figure 3). Boole's rule could have an obviously better result if the function had a lot of ups and downs, but in our case, that doesn't happen. If we use it, it will add complexity without more accuracy in our problem. The Simpson's rule 1/3 is shown below in equations 6, 7 and 8. However, we have to remember that it can work only if the number of intervals (n) is even (2, 4, 6, …). As we can see in equation 6, Simpson's rule 1/3 is written only for simple integral. However, we can reduce a double integral to just two simple integrals equations (9-13), and then we can use Simpson's 1/3 for each of them.

$$\int_a^b f(x)\,dx = \frac{h}{3}\sum\nolimits_{i=0,2,4,\ldots}^{n} [f(x_i) + 4f(x_{i+1}) + f(x_{i+2})] \tag{6}$$

$$h = (b-a)/n \tag{7}$$

$$x_i = a + i * h, \qquad i = 0,1,2,\ldots,n \tag{8}$$

$$\int_{low}^{up}\int_{g(x)}^{h(x)} f(x,y)\,dy\,dx = \int_{low}^{up}\frac{h_y(x)}{3}\sum_{i=0,2,4\ldots}^{n_y-2}[f(x,y_i)+4f(x,y_{i+1})+f(x,y_{i+2})]\,dx \tag{9}$$

$$h_y(x) = \frac{h(x)-g(x)}{n_y} \tag{10}$$

$$F(x) = \frac{h_y(x)}{3}\sum_{i=0,2,4\ldots}^{n_y-2}[f(x,y_i)+4f(x,y_{i+1})+f(x,y_{i+2})] \tag{11}$$

$$\int_{low}^{up} F(x)\,dx = \frac{h_x}{3}\sum_{j=0,2,4\ldots}^{n_x-2}\left[F(x_j)+4F(x_{j+1})+F(x_{j+2})\right] \tag{12}$$

$$h_x = \frac{up-low}{n_x} \tag{13}$$

The algorithm steps:

1. *Calculate h_x (equation 13).*
2. *For $j = 0, 2, 4, \ldots, n_x - 2$.*
    - i. *Calculate x_j, x_{j+1}, x_{j+2} (equation 8).*
    - ii. *Calculate $h_y(x_j)$, $h_y(x_{j+1})$, $h_y(x_{j+2})$ (equation 10).*
    - iii. *For $i = 0, 2, 4, \ldots, n_y - 2$.*
        - a. *Calculate $y_i(x)$, $y_{i+1}(x)$, $y_{i+2}(x)$ (equation 8) for each $x = x_j, x_{j+1}, x_{j+2}$.*
        - b. *Calculate $F(x)$ (equation 11) for each $x = x_j, x_{j+1}, x_{j+2}$.*
    - iv. *Calculate I_j (equation 12).*
3. *Calculate $I_{tot} = I_0 + I_2 + \ldots + I_{n-2}$.*

It’s important to note that for simplicity in our calculation, n_y will be equal to n_x ($n_y = n_x$), but this is not necessary. n_y can be constant ($n_y = c \neq n_x$) or a function of x ($n_y = n_y(x)$) and that can be a good idea. However, we have to be careful about which function we will pick for $n_y(x)$ because, as we mentioned earlier, Simpson's 1/3 rule only works if $n_y(x) = 2, 4, \ldots$ for each x.

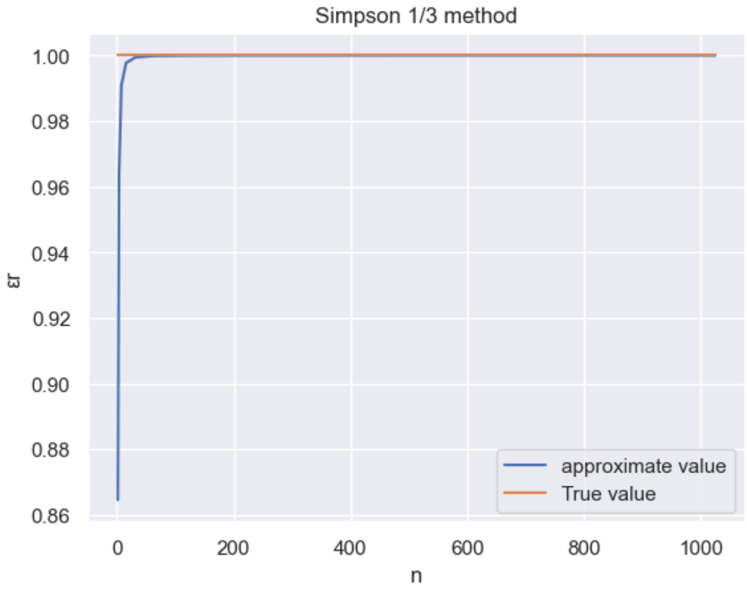

Figure 4 In this figure, we can see how Simpson's 1/3 rule approximates the real value as "n" increases (where n_x = n_y = n).

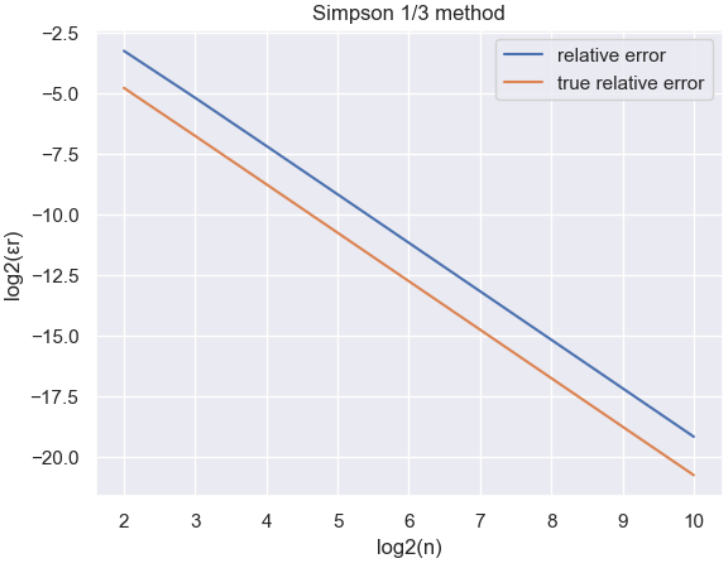

Figure 5 In this figure, we can observe how the logarithm of relative error and true relative error of Simpson's 1/3 rule decrease as logarithm of "n" increases (where n_x = n_y = n).

It is hard to approach the precision of the method theoretically by the value of n because we have a double integral when we use the Simpson 1/3 rule. It is even harder in our case because the first integral has limits which are functions of x, and that affects the h_y which is dependent on both n and x. What we can say by observing Figure 4 is that for an n > 100 (in our case only, we are not generalizing), the method increases the running time without proportional precision.

In Figure 5, we can see something incredible, the logarithm of the relative error is linearly decreasing with respect to the logarithm of n (at least for our data). We are not going to explore this in this assignment, but we could find the linear dependence of these values using the simple linear regression method. Then, if we performed some operations, we would get a recursive form of the integral value which is only dependent on a value of the integral for a value of n (n=n$_0$). The reason why we are not going to do this, is that these findings can’t be generalized for each function f and for different limits of the variables.

The difference between the relative error and the true relative error is absolutely logical and it is due to the way that we are calculating the relative error, where we are using two successive integrations for the convergence.

Monte Carlo Calculation

Monte Carlo integration is a stochastic simulation technique that is easy to implement and is based on geometric proportions. The main idea of this technique is that we can generate random points within a rectangle with uniform distribution that encompasses the volume of the function we want to integrate (as shown in Figure 6). We then count the points that fall within the volume of integration. After that, using Equation 14, we can easily evaluate the integral. The major advantage of this technique is that it is easy to implement in higher dimensions.

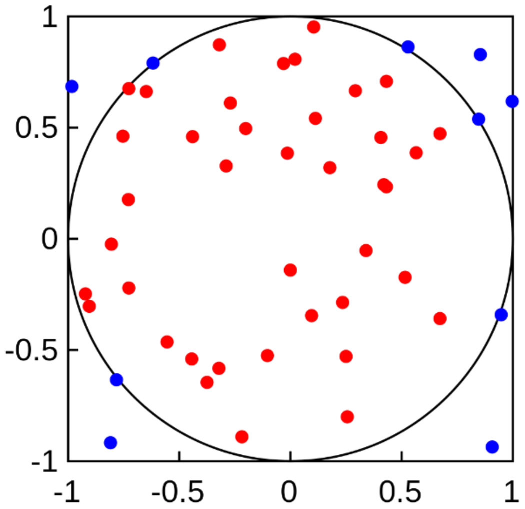

Figure 6 A representation of the use of the Monte Carlo integration technique for calculating the area of a circle inscribed in a square.

$$I = \frac{vector\ in}{total\ vectors} * I_{rectangle} \tag{14}$$

Based on what we discussed earlier, the question is: how do we determine the space where we produce the random points that include integration (2)? Because we have a double integral, we need two dimensions for the variables (x, y) and one more for the function value f(x, y). With these data, we need a 3-dimensional shape, and because we need to integrate this shape (14), the shape has to be simple to calculate its integration. The easiest shape for our task is the cuboid. Now, the only thing we have to do is to calculate its dimensions. As we mentioned earlier, the only condition for the shape that we will choose is to include the integral volume, and the technique will then theoretically converge to the true value as the number of random points tends to infinity. But that is a big problem because we can't wait forever for the method to give us a good approximation. To optimize this, we need the smallest possible cuboid that can include all the volume of the integral in the integration area. For the x-axis of the cuboid, things are simple because the limits of the x variable integration give us the coordinates for the length, which are [0, 1]. Now we have to determine the coordinate of the y-axis of the cuboid, but the problem is that we cannot use the limits of y variable integration directly as before because they are functions of x. What we do is to find the minimum and maximum value of the functions [x, 2x] for x∈ [0, 1], and the answer will be the coordinates of the y-axis of the cuboid. In our case, the answer is [0, 2]. The last coordinates we need are for the z-axis of the cuboid, which represents the value of the f function. Because all integrals are calculated from zero to the function value, we already have the lower coordinate of the z-axis, which is zero. For the upper coordinate, we just need the maximum value of f(x, y) for x∈ [0, 1] and y∈ [0, 2]. Because the function is increasing for both x and y, the answer is f(1, 2) = 9. The coordinates of our cuboid for each axis are: x [0, 1], y [0, 2], and z [0, 9]. For x and y the coordinates agree with the Figure 3.

The algorithm steps:

1. Repeat from 2 to 5 N times.
2. Generate a random number from a uniform distribution in the range of [0, 1] and set it as rx.
3. Generate a random number from a uniform distribution in the range of [0, 2] and set it as ry.
4. Generate a random number from a uniform distribution in the range of [0, 9] and set it as rz.
5. Check if rx < ry < 2*rx and rz < f(rx, ry) are both true. If they are, this means that the point is in the integration volume, so increment the success count by 1.
6. Results = 1*2*9*successes/N (equation 14).

In Figure 7, we can see the results of the technique along with their standard deviations for different values of N on a logarithmic scale. As we mentioned above, an increase in N provides a better approximation to the true value. This is evidenced by the standard deviation bars, which get smaller as N increases.

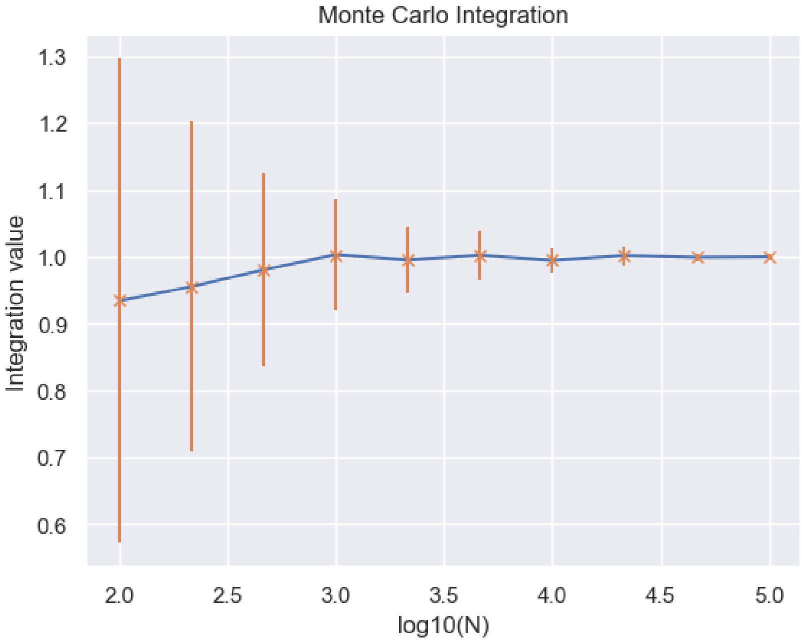

Figure 7 In this graph, we can see how the Monte Carlo method approaches the true value and how the standard deviation decreases as the number of random points increases. Standard deviations are calculation with 100 simulations for each N.

Conclusion

The results of both methods are very good, but in terms of performance, Simpson's 1/3 rule provides better accuracy for less effort. In our case, it requires fewer than 100 repetitions to obtain a very good approximation of the true value (Figure 4). On the other hand, Monte Carlo needs more than 1000 repetitions to achieve a good approximation, but it still has a large variance. As with every computational method, there are trade-offs: Monte Carlo isn't as fast as Simpson's 1/3 rule, but it is easier to code and to scale up to higher dimensions. Finally, it's important to emphasize that both methods can utilize parallel computing because their repetitions are independent of each other, which can significantly improve performance.

References

Burden, Richard L. and J. Douglas Faires. 2011. *Numerical Analysis.* s.l. : Richard Stratton, 2011.

Chapra, Steven C. and Raymond P. Canale. 2015. *Numerical Methods for Engineers.* SEVENTH EDITION. United States of America : McGraw-Hill, 2015.