Level 5 Computational Physics Exercise 2

The deadline for this exercise is Sunday 11th November 2012 at midnight. Your report and all program files should be uploaded into Blackboard at the appropriate point in the Computational Physics (PHYS2COMP) course.

S. Hanna

Objectives of the exercise

- To experiment with numerical differentiation and explore its accuracy;
- To experiment with Simpson's rule for numerical integration.
- To explore the use of Monte Carlo methods for numerical integration.

We expect that you will need help as you develop your programs. Please consult the demonstrators as frequently as you need to during the drop-in sessions. They are there to help.

Report

As previously, you should prepare a concise report outlining your methods and highlighting your findings with suitable graphs or tables. Credit will be given for a reasoned discussion of your findings.

Problem 1: Numerical Differentiation – First Derivative (8/20 marks)

We may obtain a numerical estimate of the derivative of a function using the forward / backward or central difference formulae shown in lectures:

$$f'_f(x) = \frac{f(x+h) - f(x)}{h} \quad ; \quad f'_c(x) = \frac{f(x+h) - f(x-h)}{2h}$$

The smaller the value of h , the better the approximation. However, h cannot be made too small, or rounding errors will reduce the precision of the calculation. For the forward difference formula, the optimum value is suggested to be $h = x\epsilon^{1/2}$ where ϵ is the relative precision of the floating-point format i.e. roughly 1 part in 10^7 for single precision and 1 part in 10^{15} for double precision variables.

a) Working in either single or double precision, write a program which is able to find the first derivative of $f(x) = \sin(x)$, at a given value of x, for a given h, using both the forward and central difference formulae. Test your routine to make sure it works as expected.

b) Using a loop in your program, output values of the first derivative of $\sin(x)$ in the range $0 \leq x \leq 2\pi$. Plot the values on a graph and compare your result with the analytic value for different values of h.

   HINT: get your routine to write out to a file, columns of "$x, f(x), f'_f(x), f'_c(x), f'(x) - f'_f(x), f'(x) - f'_c(x)$", where $f'(x) = \cos(x)$ is the analytical result. Load this into Excel, or your favourite plotting program, to produce the graphs.

c) For a fixed value $x = \pi/4$, plot the modulus of the error in $f'_f(x)$ and $f'_c(x)$ as a function of h. Hence determine the optimal value of h for forward and central differences. How much more accurate is the central difference formula?

   HINT: for single precision take $10^{-1} \leq h \leq 10^{-9}$, and for double precision take $10^{-1} \leq h \leq 10^{-18}$. Plot $\log(|error|)$ versus $\log(h)$.

Problem 2: Numerical Integration – Simpson's Rule (6/20 marks)

In your first year lectures, you used the small angle approximation to find the time period of a simple pendulum as a function of its length:

$$T = 2\pi\sqrt{\frac{l}{g}}$$

You were told that the equations of motion became impossible to solve if the angle of oscillation was too large. In fact, the equation of motion reduces to:

$$T = 4\sqrt{\frac{l}{2g}}\int_0^{\theta_0}\frac{d\theta}{\sqrt{\cos\theta - \cos\theta_0}}$$

where θ_0 is the angle at $t = 0$ when the pendulum is first released.

We now have the tools to deal with this. However, if you plot this function, you will find that you cannot integrate it using the trapezium rule or Simpson's rule, because there is a singularity at the upper limit.

To avoid this problem, we introduce a new variable ψ:

$$\sin(\theta/2) = \sin(\theta_0/2)\sin\psi$$

and make use of a trig formula for the difference of two cosines, to obtain (after some tricky manipulation):

$$T = 4\sqrt{\frac{l}{g}}\int_0^{\pi/2}\frac{d\psi}{\sqrt{1-\sin^2(\theta_0/2)\sin^2\psi}} \tag{1}$$

which works for any $\theta_0 < \pi$. This is an example of an *elliptic integral of the first kind*.

a) Evaluate Eqn. (1) numerically using Simpson's rule (see lectures).

b) Plot the time period of the pendulum, T, for $0 \le \theta_0 < \pi$. Make sure your calculation converges to the expected value for small angles. Repeat your plot for different numbers of points, N, in the Simpson's rule expression, and comment on the effect of N.

   HINT: use a loop for θ_0 and write your values of $T(\theta_0)$ to a file.

c) For what value of θ_0 does T vary from the small angle limit by 10%?

Problem 3: Monte Carlo integration – calculating π (6/20 marks)

Some integrals are difficult to solve efficiently, either analytically or using numerical integration. As noted in the lectures, there is an alternative approach, which relies on the use of random numbers to 'sample' a function across its domain in order to estimate the integral. This approach (called Monte Carlo integration after the famous gambling centre) has advantages and disadvantages compared to standard numerical methods. In some cases, most notably integrals over oddly-shaped regions, or in higher dimensions, it can provide a staggering increase in efficiency.

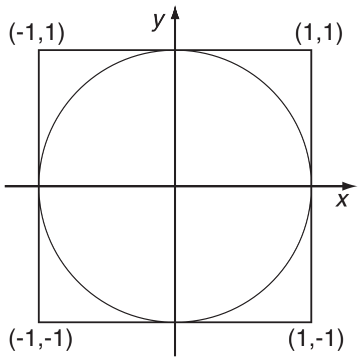
The Monte Carlo method is captured in the following equation:

$$\int f\,dV \approx V\left(\langle f\rangle \pm \sqrt{\frac{\langle f^2\rangle - \langle f\rangle^2}{N}}\right) \tag{2}$$

where angle brackets denote the average value over a set of N random sample points of the function. Quite often, we will wish to integrate over a volume which is oddly shaped (e.g. the integration limits in one dimension are a function of those in another). This is OK, since we can choose sample points from a regularly-shaped volume which includes the integration region, and then throw away the points we do not need i.e. we assign the function the value zero for these.

As an example of how to estimate a quantity using random numbers, you will write a program to evaluate π. Conceptually, we draw a square and inscribe a circle within it (see figure). If we choose points within the square at random, we should find that the ratio of the number of points lying within the circle to those outside is:

$$\frac{n_{\text{inside}}}{n_{\text{total}}} \approx \frac{\pi}{4}$$



In the context of Eq. (2) you are performing the integral:

$$\int_{-1}^{1}\int_{-1}^{1} f(x,y)\,dxdy \quad \begin{cases} f=1 & \left|x^2+y^2\right| \le 1 \\ f=0 & \left|x^2+y^2\right| > 1 \end{cases} \tag{3}$$

In practice, you need only count the points.

a) Write a program to evaluate the integral in Eqn. (3) using the Monte Carlo approach.

b) Investigate the convergence of your integral on the ‘correct’ value of π as a function of the number of sample points used, N. From Eqn. (2) you would expect your value of π to improve indefinitely with increasing N. Is this what you find?

Monte Carlo integration depends on reliable random numbers. You should read about random number generators in a textbook such as *Numerical Recipes*, and outline their potential deficiencies in your report.

Submitting your work

You should submit the following to Blackboard:

1. A concise report, in MS Word or pdf format;
2. Your program for numerical differentiation, as used for Problem 1;
3. Your program for numerical integration using Simpson’s rule, as used for Problem 2.
4. Your program for numerical integration using Monte Carlo methods, as used for Problem 3.

Please note the following points:

- Blackboard will not accept your compiled program—please only upload your “prog.c" files.
- Blackboard (Turnitin) also won’t accept a file ending “.c" so please rename your file with a “.txt" extension.
- Please also give your programs sensible distinguishing names, including your name or userid e.g. “my_userid_ex2_prob_1.txt" or “myname_ex2_prob_3.txt".

If you have any problems submitting your work, please contact Dr. Hanna (s.hanna@bristol.ac.uk) or ask a demonstrator.