Computational Project Description

Summary

The aims of the Project are to describe methods for solving given computational problems, develop and test MATLAB code implementing the methods, and demonstrate application of the code to solving a specific computational problem. In this Project, you will be required to demonstrate

1. ability to investigate a topic through guided independent research, using resources available on the internet and/or in the library;
2. understanding of the researched material;
3. implementation of the described methods in MATLAB;
4. use of the implemented methods on test examples;
5. ability to present the studied topic and your computations in a written Project Report.

Plagiarism and Declaration

1. This report should be your independent work. You should not seek help from other students or provide such help to other students. All sources you used in preparing your report should be listed in the References section at the end of your report and referred to as necessary throughout the report.

2. Your Project Report must contain the following Declaration (after the title page):

DECLARATION

All sentences or passages quoted in this Project Report from other people's work have been specifically acknowledged by clear and specific cross referencing to author, work and page(s), or website link. I understand that failure to do so amounts to plagiarism and will be considered grounds for failure in this module and the degree as a whole.

Name:

Signed: (name, if submitted electronically)

Date:

Project report

The report should be about 8-10 pages long, written in Word or LaTeX. Equations should be properly formatted (use Insert Equation in Word) and cross-referenced, if necessary. The report should be submitted via Blackboard in a single file (Word document or Adobe PDF) and contain answers to the tasks given below. Tasks 1 and 2 carry 5 marks each while Task 3 carrys 10 marks.

Topic 1: Numerical Differentiation

Consider a function $f(x)$ of your choice and a point $x = a$ in its domain.

Task 1: Derive second order accurate forward, backward and centered finite difference schemes for finding the derivative $f'(x)$ at $x = a$.

Task 2: Derive second order accurate forward, backward and centered finite difference schemes for $f''(x)$ evaluated at $x = a$.

Task 3: Write a script file which plots $f'(x)$ and $f''(x)$ vs x using the schemes you derived in (a). Make sure you use a centered scheme at interior points, forward finite difference at the left end point and backward difference at the right end point of your numerical grid.

Topic 2: Numerical Integration

In lectures, you learnt the Trapezoidal rule to evaluate integrals when the integrand was a function of a single variable x. We can extend this idea to two dimensions to evaluate double integrals of the form

$$I = \int_c^d \int_a^b f(x, y) dx dy. \tag{1}$$

where the integration is performed on the rectangle $R = \{(x, y) \in \mathbb{R}^2 : a \leq x \leq b$ and $c \leq y \leq d\}$. Create a 2D numerical grid such with equal spacing in each dimension. This can be done by creating two grids one for each dimension. Now, follow the steps below to evaluate the integral I.

Task 1: Keeping the variable y fixed, apply trapezoidal rule to calculate the integral with respect to x.

$$g(y) = \int_a^b f(x, y) dx \tag{2}$$

Task 2: Apply trapezoidal rule to integrate $g(y)$ with respect to y. This gives

$$I = \int_c^d g(y) dy \tag{3}$$

Deduce a formula for I in the form of a double sum.

Task 3: Write a function in MATLAB which uses your derived formula to evaluate I for a given function $f(x, y)$ and limits a, b, c and d of your choice. Compare your value of I with the value given by MATLAB's built-in function integral2().