CAS708/CSE700 Scientific Computing

Ned Nedialkov

Winter 2024

E-mail: nedialk@mcmaster.ca
Office Hours: TBD
Office: ITB 123

Web: http://www.cas.mcmaster.ca/~nedialk/
Class Hours: Mo/Fr 9:00–10:30
Class Room: ITB 222

Overview

This course is a graduate-level introduction to the principles and methods of scientific computing. Designed for students with a strong foundation in linear algebra and calculus, the course aims to deepen understanding of numerical methods and their applications.

Through a combination of theoretical instruction and practical exercises, students will explore the intricacies of floating-point computing, error analysis, and the development and implementation of fundamental numerical algorithms.

Required Materials

- Required textbook
  Uri Ascher and Chen Greif. A First Course in Numerical Methods, Society for Industrial and Applied Mathematics (SIAM) 2011.
- Course notes will be available on Avenue

Prerequisites

A strong foundation in linear algebra and calculus is essential. It's expected that students have some prior knowledge in numerical methods. Basic methods such as Gaussian elimination and polynomial interpolation, usually taught in undergraduate courses, will not be extensively reviewed. This course will extensively utilize Matlab.

Course Objectives

The students in this course should learn

- issues in floating-point computations and roundoff error analysis
- fundamental numerical algorithms

- techniques for analyzing numerical algorithms

and should be able to

- analyze numerical algorithms
- implement numerical algorithms efficiently
- interpret numerical results
- choose the appropriate numerical method for a particular problem

Course Content

This course will cover the following topics:

- Errors in scientific computing, sensitivity and conditioning, algorithm stability
- Floating-point arithmetic, roundoff error analysis
- Linear systems; direct and iterative methods
- Linear least squares; normal equations, orthogonal transformations, QR
- Eigenvalues and singular values
- Nonlinear systems
- Piecewise polynomial interpolation
- Numerical and automatic differentiation (AD)
- Initial value problems in ordinary differential equations (ODEs)
- Introduction to deep learning

Assessments

The course assessment will consist of four assignments (totaling 40%) and a project (60%). The assignments will consist of a mix of theoretical questions, to be answered with pencil-and-paper, and practical tasks requiring implementation in Matlab.

Assignment	weight	dates	topics
1	10%	29 Jan–9 Feb	floating-point, linear systems
2	10%	9–23 Feb	least-squares, SVD, QR
3	10%	1–15 March	interpolation, nonlinear systems, AD
4	10%	16–30 March	ML, ODEs
Project proposal	–	10 March	
Project	60%	10 April	

The above dates may change depending on how we progress with the material. Assignment grades will be indicated on the marked hard copies of the assignments.

Project. The final project should include concepts studied in this course and scientific computing in general. It should present relevant theory and include software and numerical results.

The write up should be of the quality of a technical report (at least). Marks will be deducted for poor explanations, typos, and sloppiness in general. Your programs should be well structured, explained and properly documented. When executed, your code should produce the relevant numerical results and figures.

The written reports must be typeset in LaTex in 12pt, single space, “fullpage." For this purpose, you should put the following two lines at the beginning of your LaTeX file

```
\documentclass[12pt]{article}
\usepackage{fullpage}
```

You can work in groups of at most three. The expected length is ≥ 8 pages per person in a group. This page length includes text and numerical results, but not the programs.

Some Resources

- W. Cheney and D. Kincaid. Numerical Mathematics and Computing. Brooks/Cole.
- G. Dahlquist and A. Björck. Numerical Methods in Scientific Computing. SIAM, 2008.
- M. Heath. Scientific Computing. McGraw-Hill 2002
- N. Higham. Accuracy and Stability of Numerical Algorithms. SIAM.
- G. W. Steward. Afternotes on Numerical Analysis. SIAM, 1996.
- C. van Loan. Introduction to Scientific Computing. A Matrix-Vector Approach using MATLAB, Prentice-Hall, 2000.
- T. Kelly Iterative Methods for Linear and Nonlinear Equations
- D. Goldberg. What Every Computer Scientist Should Know About Floating-Point Arithmetic. HTML http://docs.oracle.com/cd/E19957-01/806-3568/ncg_goldberg.html PDF http://www.validlab.com/goldberg/paper.pdf
- Short introduction to floating-point numbers by Clever Moler https://www.mathworks.com/company/newsletters/news_notes/pdf/Fall96Cleve.pdf

Course Policies

Changes

The instructor reserves the right to modify elements of this course and will notify the students accordingly.

Attendance

It is desirable that you attend all the lectures. However, your grade will not be affected if you miss a lecture.

Late Assignments

A late assignment will receive a 10% penalty per day.

Academic Integrity and Honesty

You are expected to exhibit honesty and use ethical behaviour in all aspects of the learning process. Academic credentials you earn are rooted in principles of honesty and academic integrity.

Academic dishonesty is to knowingly act or fail to act in a way that results or could result in unearned academic credit or advantage. This behaviour can result in serious consequences,

e.g. the grade of zero on an assignment, loss of credit with a notation on the transcript (notation reads: 'Grade of F assigned for academic dishonesty'), and/or suspension or expulsion from the university.

It is your responsibility to understand what constitutes academic dishonesty. For information on the various types of academic dishonesty please refer to the Academic Integrity Policy, located at http://www.mcmaster.ca/academicintegrity

The following illustrates only three forms of academic dishonesty:

1. Plagiarism, e.g. the submission of work that is not one's own or for which other credit has been obtained.
2. Improper collaboration in group work.
3. Copying or using unauthorized aids in tests and examinations.

Authenticity/Plagiarism detection. Some courses may use a web-based service (Turnitin.com) to reveal authenticity and ownership of student submitted work. For courses using such software, students will be expected to submit their work electronically either directly to Turnitin.com or via an online learning platform (e.g. Avenue to Learn, etc.) using plagiarism detection (a service supported by Turnitin.com) so it can be checked for academic dishonesty. Students who do not wish their work to be submitted through the plagiarism detection software must inform the Instructor before the assignment is due. No penalty will be assigned to a student who does not submit work to the plagiarism detection software. All submitted work is subject to normal verification that standards of academic integrity have been upheld (e.g., on-line search, other software, etc.). For more details about McMaster's use of please go to Turnitin.com.

Accommodations for Disabilities

Students who require academic accommodation must contact Student Accessibility Services (SAS) to make arrangements with a Program Coordinator. Academic accommodations must be arranged for each term of study. Student Accessibility Services can be contacted by phone 905-525-9140, ext. 2865 or e-mail sas@mcmaster.ca. For further information, consult McMaster University's Policy for

Faculty Notices

"The Faculty of Engineering is concerned with ensuring an environment that is free of all discrimination. If there is a problem, individuals are reminded that they should contact the Department Chair, the Sexual Harassment Officer or the Human Rights Consultant, as the problem occurs."