Numerical Methods I, Fall 2014

1 Logistics

Course webpage
http://cims.nyu.edu/~donev/Teaching/NMI-Fall2014
including links to textbook and software, assignments, and lecture slides

Registration
3pts per semester, MATH-GA 2010.001 / CSCI-GA 2420-001

Instructor
Aleksandar Donev, http://cims.nyu.edu/~donev
Office: 1016 Warren Weaver Hall
E-mail: donev@courant.nyu.edu ; Phone: (212) 992-7315

Grader
Not yet assigned

Lectures:
5:10 - 7 pm Thursdays, 101 Warren Weaver Hall (Courant Institute)

Office Hours:
3 - 5 pm Tuesdays to be discussed on first class, or by appointment

Communication
There is a message board on the NYU Courses webpage that will be used for
messages related to the assignments and any scheduling changes, as well as to post
grades (for privacy) and to submit homeworks.

2 Content

This course is part of a two-course series meant to introduce Ph.D. students in mathematics to the
fundamentals of numerical mathematics (but any Ph.D. student seriously interested in applied mathematics
should take it). It will be more technical (theoretical) than the Scientific Computing course, which is meant
for masters students and covers similar material as this two-course series but in less detail. This course will
not cover differential equations, which form the core of the second part of this series, Numerical Methods II.

2.1 Recommended Textbooks

The course will be based on my lecture notes and homework assignments. It will not follow any specific
text. Nevertheless, it is extremely important for you to read other texts to get a different and broader
perspective on the field. The following books are recommended reading and are easily available via
Courant’s library including electronic copies, linked on the course homepage:

NY: Springer-Verlag.
• Corless, R.M., & Fillion, N. (2014). A Graduate Introduction to Numerical Methods: From the
Industrial & Applied Mathematics. (Available on reserve at the Courant Library.)

2.2 Prerequisites

A good background in linear algebra, and experience with writing computer programs (in Matlab, Python,
Fortran, C, C++, or other language). Prior knowledge of Matlab is not required, but it will be used
as the main language for the course. If you have experience with compiled languages (Fortran, C, C++),
Matlab will be easy to learn and use, and comes with a great help facility.

You will not be required to learn and use a compiled language, especially if you are relatively new
to computing, but I will strongly encourage you to and assist you in the learning process (I am a Fortran
expert but will do my best to help you with other languages as well).
2.3 Assignments and Grading
There will be regular assignments and a final project. The assignments will be mostly computational but some will be theoretical problems, and are the most important part of this course. There will be a final take-home exam examining a topic of relevance not covered in the class. The grade will be based 60% on the homeworks and 40% on the final.

2.4 Computing
Computing on your own will form an essential part of the learning process and you will be exposed to Linux, Matlab, compiled languages, LaTeX, and other tools commonly used by computational scientists. The Courant Institute has computer labs with Linux workstations that have Matlab (matlab), Maple (xmaple), Mathematica (mathematica), the GNU (gcc, g++, gfortran), Intel and Pathscale C/C++/Fortran compiler suites, and other useful software installed.

3 Plan of lectures
This is only a tentative list for the lectures; consult the webpage for the up-to-date information.

**Recommended preparation** (before first class!): Start playing with MATLAB.

- Numerical Computing: Well posedness, numerical conditioning, stability, floating point calculations (IEEE, roundoff, truncation errors), scientific software (MATLAB, compiled languages, GPUs, etc).
- Direct methods for linear systems: Gaussian elimination, LU, Cholesky factorization, pivoting, dense and banded systems.
- Dense eigenvalue problems: diagonalization, and QR and SVD decompositions.
- Iterative methods for eigenvalues: power method, spectral radius and convergence of iterative schemes. Applications of eigenvalue methods such as principal component analysis.
- Nonlinear equations: One dimensional root finding (bisection, secant), fixed-point iteration, Newton’s method, safeguarded Newton, Aitken acceleration, higher dimensions.
- Interpolation: One dimensional polynomial interpolation (Newton, Lagrange), extensions to two and three dimensions. Splines (cubic and B-splines).
- Orthogonal polynomials: Interpolation (Chebyshev, Legendre, Gauss), approximation. Least squares approximation.
- Spectral approximation and accuracy. Orthogonal trigonometric polynomials and Discrete Fourier Transforms (DFT).
- Fast Fourier transforms (FFT) and wavelets. Convolution. Filtering/smoothing. Numerical differentiation via finite differences and spectral methods.
- Numerical integration: One-dimensional quadrature (midpoint, trapezoidal, Simpson, Newton-Cotes, Hermite methods, etc.), Richardson extrapolation, automatic integration, extensions to two and three dimensions and the curse of dimensionality.
- Monte Carlo Methods: Pseudo-random numbers (uniform, normal, and other distributions), numerical integration in high dimensions, sampling variance, and variance reduction.