FACULTY OF SCIENCE

SCHOOL OF MATHEMATICS
AND STATISTICS

MATH3311

Mathematical Computing for Finance

MATH5335

Computational Methods for Finance

Term 2, 2019

Cricos Provider Code: 00098G

MATH3311/MATH5335 – Course Outline

Information about the course

Course Authority: Prof Josef Dick, RC-2074, email `josef.dick@unsw.edu.au`

Consultation: Check UNSW Moodle or by email

Credit, Prerequisites, Exclusions:

These courses count for 6 Units of Credit (6 UOC). MATH3311 prerequisites are

- MATH2121 or MATH2221 or MATH2120 or MATH2130 Mathematical Methods for Differential Equations
- MATH2501 or MATH2601 Linear Algebra
- 6 UOC of Level 2 Statistics.

In addition, it is beneficial to have prior experience with MATLAB and with simple procedural programming, such as is provided in MATH2301 Mathematical Computing or MATH1151/MATH1251 Mathematics for Actuarial Studies and Finance.

There is no higher version of MATH3311.

The postgraduate course MATH5335 has no formal prerequisites but students need to have taken courses roughly equivalent to those mentioned above. MATH3311 and MATH5335 exclude each other.

Classes: Combined Lectures (3 hours) for MATH3311 and MATH5335

Time and Lecture Room:

Monday	17–19	Electrical Engineering G23	Weeks 1, 3–5, 7–10
Wednesday	17–18	Red Centre Theatre	Weeks 1–5, 7–9

(Monday Week 2 is a public holiday; We will have another lecture on Monday in Week 10;)

MATH3311 has two hours of laboratory/tutorial classes taught by Yi-Lung Chen

Friday	12–14	Red Centre G12C	Weeks 1–10

MATH5335 has a one hour laboratory/tutorial class: There are two streams for this tutorial: the first stream is taught by Josef Dick

Monday	19–20	Red Centre G12A	Weeks 1, 3–10

and a second stream taught by Yi-Lung Chen

Wednesday	18–19	Red Centre G12A	Weeks 1–10

In Week 9, the computing laboratory test will be held in place of the normal tutorial for Math3311 and on Wednesday from 18.10–19.40 for Math5335.

Students in the graduate course MATH5335 are expected to display much more independence, especially in working through the laboratory exercises.

UNSW Moodle: Further information, skeleton lecture notes, sample programs and other material will be provided via Moodle.

`https://moodle.telt.unsw.edu.au`

You should log on to the MATH3311/MATH5335 Moodle course **regularly** to check for announcements and new material.

Course aims

Ultimately finance is concerned with making definite numerical recommendations which frequently can only be obtained by analysing sophisticated models using high-speed computers. This course studies the design, implementation and use of computer programs to solve practical mathematical problems of relevance to finance, insurance and risk management.

Relation to other mathematics courses

The most closely related courses are MATH2301 *Mathematical Computing* and MATH3101 *Computational Mathematics* (with its postgraduate version MATH5305). Some aspects are also related to the Numerical Methods component of MATH2089 *Numerical Methods and Statistics*, which is only available to students in Engineering programs.

Student Learning Outcomes

Practical solution of problems of relevance to finance using Matlab is an essential component of this course. This prepares you for the finance/insurance workforce where software like Matlab or Excel/VBA is used for research, investigation and proof of concept before it is passed on for professional programming, often in C/C++, integrating with the company's front end systems, databases and risk management processes.

Relation to graduate attributes

This course will help develop your quantitative skills and ability to reason logically and mathematically and apply these skills to problems of relevance to finance, insurance and risk management. It will emphasise the importance of understanding the underlying mathematics, computational techniques and problems solving skills. It will help establish what confidence you should have in the computed answers and what are the risks associated with a given computational problem.

You are expected to always think about your results and their consequences, not just blindly believe what is produced by a computer.

The above outcomes are related to the development of the Science Faculty Graduate Attributes, in particular:

1. **Research, inquiry and analytical thinking abilities**,

4. **Communication**,

6. **Information literacy**

Teaching strategies underpinning the course

The lectures will introduce a wide range of computational methods used in a variety of financial applications. During the lab classes, you will use MATLAB to study examples involving these methods. The examples will illustrate key features of the methods, including any significant limitations.

The key to success in MATH3311/MATH5335 is to work through as many of the lab exercises as possible and to understand what is happening.

Assessment

Summative Assessments

The course has four in semester assessment tasks plus a final exam, weighted as follows:

Assignment	20%
Lab Test (Week 9)	20%
Final Exam (2 hours)	60%

The Assignment is an ongoing assessment which is comprised of four parts. The due dates will be announced on the course Moodle page.

It can be done in groups comprised of $1, 2, 3$ or 4 members. The contribution of each member must be specified (as a percentage).

The Lab Test is done in one of the computer laboratories using the Linux environment. It is not possible to use the Windows environment during the Lab Test.

Self-Paced Matlab lessons

There are MATLAB self-paced lessons available through Moodle and associated MapleTA quizzes. If you are not familiar with MATLAB or have not used MATLAB for some time, then these lessons should be done as soon as possible. The Maple TA quizzes on MATLAB do **not** count towards your grade, but can be used both to check your knowledge of MATLAB and to become familiar with MapleTA.

Knowledge and abilities assessed

The assignments will have a written and a computing component, and will emphasise basic MATLAB programming and will also provide a preview of topics to be covered during lectures later in the course. This task will test your skill at synthesising mathematical theory and practical computation, and at interpreting and presenting numerical results.

The laboratory test in week 9 will take place in the computer lab under exam conditions. You will have to write several short MATLAB scripts and functions to solve simpler versions of the lab exercises.

The Final Exam will mainly test your mathematical understanding of the numerical methods discussed in lectures, and of their application in a financial context.

Assessment criteria

For computing tasks, the main criteria will be correctness, efficiency and clarity. In other words, does the program produce the correct answers? Does the program run in a reasonable time with reasonable memory requirements? Can a human readily understand (from reading the source code) what the program does? For written tasks, you should set out your working clearly and in a logical sequence, with adequate justification for each step. Aim for the most direct answer possible.

Students in the postgraduate version will be expected to complete additional/harder sections in the assignment and final exam.

Additional resources and support

UNSW Moodle

All course materials will be available on UNSW Moodle. You should check regularly for new material, as well as for announcements about assessment tasks etc.

Tutorial and Laboratory Exercises

Exercise sets for the laboratories/tutorials will be provided on UNSW Moodle. In the labs, you will work independently at your own pace, but staff will be available for discussion and to help when necessary. Take advantage of this help: it may save you a great deal of time.

During the lecture I will outline which lab exercises you are expected to cover over the coming week. Before each lab, you should prepare by reading the relevant exercises and at least making a start on some of them.

Lecture notes

Lecture notes, including laboratory exercises, and also notes summarising key knowledge about MATLAB and statistical concepts, will be provided through Moodle.

Software

The software required for this course is available on the computers in the School of Mathematics and Statistics computer laboratories RC-G012 and RC-M020.

Additionally, the UNSW MATLAB site license allows current UNSW students to download a copy of MATLAB for use on their own computer for their courses. See

`https://www.maths.unsw.edu.au/currentstudents/obtaining-software`

The assignment **must** run in the version of MATLAB used in the School of Mathematics and Statistics computer laboratories. Both your assignment and the computing laboratory exam will be processed using this version of MATLAB.

Computer laboratories

See the School of Mathematics and Statistics web page for information about opening times etc.

Remote access Linux Lab PC

On UNSW Moodle you will find information on remote access to servers which can be used to run mathematical and statistical software such as MATLAB, Maple and R.

References

There is no textbook which covers all aspects of this course. Use the library catalogue to search the library for books on *Computational Finance*, *Mathematical Computing*, *Mathematical Software*, MATLAB, etc. A book which follows the themes of this course is *Numerical Methods in Finance: A Matlab based introduction* by Paolo Brandimarte [2].

Standard texts on financial derivatives are Hull [11] and Wilmott [25, 26], while Glasserman [6] and Jäckel [12] are good for Monte-Carlo based simulation methods. References [24, 14, 23] are more general texts on computational aspects of finance.

General References

Numerical methods for partial differential equations [15, 18, 22] are a key part of the subject. Methods based on Fourier analysis [3] are also important. A general reference on applied mathematics covering many aspects of this subject is Strang [20]. General references on scientific computing include [4, 7, 13]. Nash [17] covers the history of scientific computing.

Linear Algebra

An elementary treatment of linear algebra can be found in Strang [21], while a more advanced reference is Golub and Van Loan [8]. The LAPACK manual [1] contains both descriptions of the methods and available software.

Mathematical Software

Solving practical problems typically requires programming, either in a high level numerical language like Matlab [10], a symbolic package like Maple, Mathematica or MuPad, or a lower level language like Fortran or C. If you are using Fortran or C the routines available in the Numerical Recipes [19] book, the LAPACK library [1] and the NAG Library [16] may be very useful. The Intel Math Kernel Library (MKL) contains multi-threaded versions of the BLAS (Basic Linear Algebra Subroutines), LAPACK and other useful routines.

References

[1] E. Anderson, Z. Bai, C. Bischoff, J. Demmel, J. J. Dongarra, J. D. Croz, A. Greenbaum, S. Hammarling, A. McKenney, S. Ostrouchov, and D. C. Sorensen, *LAPACK Users' Guide*, SIAM, Philadelphia, third ed., 1999.

[2] P. Brandimarte, *Numerical Methods in Finance: A MATLAB based introduction*, John Wiley, New York, 2002.

[3] W. L. Briggs and V. E. Hansen, *The DFT: An Owners Manual for the Discrete Fourier Transform*, SIAM, Philadelphia, 1995.

[4] L. D. Fosdick, E. R. Jessup, C. J. C. Schauble, and G. Domik, *An Introduction to High-Performance Scientific Computing*, MIT Press, 1996.

[5] A. Gilat and V. Subramaniam, *Numerical Methods for Engineers and Scientists: An Introduction with Applications using Matlab*, Wiley, USA, 2008.

[6] P. Glasserman, *Monte Carlo Methods in Financial Engineering*, Springer, 2010.

[7] G. H. Golub and J. M. Ortega, *Scientific Computing: An Introduction to Parallel Computing*, Academic Press, New York and London, 1993.

[8] G. H. Golub and C. F. Van Loan, *Matrix Computations*, John Hopkins University Press, Baltimore and London, third ed., 1996.

[9] N. J. Higham, *Accuracy and Stability of Numerical Algorithms* , SIAM, Philadelphia, 1996.

[10] D. J. Higham and N. J. Higham, *MATLAB Guide*, SIAM, Philadelphia, 2000.

[11] J. C. Hull, *Options, Futures, and other Derivatives*, Prentice Hall, Englewood Cliffs, NJ, 8th ed., 2011.

[12] P. Jäckel, *Monte Carlo Methods in Finance*, John Wiley, Chichester, 2002.

[13] N. Knockler, *Numerical Methods and Scientific Computing: Using Software Libraries for Problem Solving*, Clarendon Press, Oxford, 1991.

[14] B. Lapeyra, A. Sulem, and D. Talay, *Understanding Numerical Analysis for Option Pricing*, Cambridge University Press, Cambridge, UK, 2000.

[15] K. W. Morton and D. F. Mayers, *Numerical Solution of Partial Differential Equations*, Cambridge University Press, Cambridge, UK, 1991.

[16] *NAG Numerical Libraries*, Numerical Algorithm Group, Oxford, U.K. http://www.nag.co.uk/numeric.html.

[17] S. G. Nash, *A History of Scientific Computing*, ACM Press, New York, 1990.

[18] J. M. Ortega and G. H. Golub, *Scientific Computing and Differential Equations: An Introduction to Numerical Methods*, Academic Press, New York and London, 1992.

[19] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes in Fortran: The Art of Scientific Computing*, Cambridge University Press, Cambridge, UK, second ed., 1992. Second edition is also available in C.

[20] G. Strang, *Introduction to Applied Mathematics*, Wellesley-Cambridge Press, Cambridge, MA, 1986.

[21] ——, *Linear Algebra and its Applications*, Harcourt Brace Jovanovich, San Diego, 3 ed., 1988.

[22] J. C. Striwerda, *Finite Difference Schemes for Partial Differential Equations*, Wadsworth and Brooks/Cole, 1989.

[23] D. Tavella, *Quantitative Methods in Derivative Pricing: An Introduction to Computational Finance*, John Wiley, Chichester, 2002.

[24] D. Tavella and K. Randall, *Pricing Financial Instruments: the finite difference method*, John Wiley, New York, 2000.

[25] P. Wilmott, *Derivatives: The Theory and Practice of Financial Engineering*, John Wiley, 1998.

[26] P. Wilmott, S. D. Howison, and J. Dewynne, *The Mathematics of Financial Derivatives: A Student Introduction*, Cambridge University Press, Cambridge, UK, 1995.

Course Evaluation and Development

The School of Mathematics and Statistics evaluates each course each time it is run. We consider the student responses and their implications for course development.

Administrative matters

It is the student's responsibility to be familiar with UNSW and School of Mathematics and Statistics policies.

Applications for Special Consideration

Please adhere to the University special consideration policy and procedures provided on the web page below. Special Consideration web site:

`https://student.unsw.edu.au/special-consideration`

For final exams with special consideration granted, the Exams Unit will email the rescheduled supplementary exam date, time and location to your student zID email account directly.

For successful applications for special consideration only.

Please ensure you regularly check your student email account (zID account) for the information on new dates to attend an assessment, or dates for any supplementary exam both in Term and Final.

The supplementary exam period/dates for the final exam can be found at this web site: `https://student.unsw.edu.au/exam-dates`

Please ensure you are aware of these dates and that you are available during this time.

School Rules and Regulations

Details of the general rules regarding attendance, release of marks, etc are available via the School of Mathematics and Statistics Web page at

`http://www.maths.unsw.edu.au/currentstudents/assessment-policies`.

Plagiarism and academic honesty

Plagiarism is the presentation of the thoughts or work of another as one's own. Issues you must be aware of regarding plagiarism and the university's policies on academic honesty and plagiarism can be found at

`http://www.lc.unsw.edu.au/plagiarism`

Detailed lecture schedule

The table below provides a rough guide for the week-by-week lecture content. This is only an estimation, we do not strictly follow this schedule.

Week	Day	Lecture	Topic
1	Mon	1, 2	Computing: hardware, software, accuracy and efficiency
	Wed	3	Vector and matrix norms
2	Wed	4	Condition numbers, sensitivity
3	Mon	5, 6	Linear systems and matrix decompositions, special matrices
	Wed	7	Data fitting: least square
4	Mon	8, 9	ℓ_1 and splines; Model calibration; Fourier analysis
	Wed	10	Discrete Fourier transform and FFT
5	Mon	11, 12	Sparse matrices: banded matrices and general sparse matrices
	Wed	13	Nonlinear equations
6			Break
7	Mon	14, 15	Implied volatility, nonlinear systems; Numerical integration
	Wed	16	Expected values: one and multi-dimensional
8	Mon	17, 18	Monte Carlo methods, pseudo-random numbers, QMC methods
	Wed	19	Weiner processes
9	Mon	20, 21	Stochastic differential equations Stochastic integrals; Option pricing
9	Wed	22	Black–Scholes formula and PDE
10	Mon	23, 24	Black–Scholes PDE: finite difference methods