University of St.Gallen

7,626: Solving Economics and Finance Problems with MatLab

Subject information

ECTS-Credits: 3

Attached courses

Timetable	Language	Lecturer
7,626,1.00 Solving Economics and Finance Problems with MatLab	**English**	**Gruber Peter**

Course information

Course prerequisites

- Knowledge of bachelor level mathematics, statistics, econometrics and finance.
- Interest in quantitative methods.
- No programming experience is required.
- This course is targeted at MiQE/F, MEcon and MBF students. All other students are welcome, but may need extra time to study the prerequisites.

Course content

Many interesting problems in economics and finance can only be solved with the help of the computer, as no analytical solutions exist. This course gives an introduction to solving problems numerically with the help of MATLAB, an easy to learn, powerful and widely used programming environment.

The course has **three** equally important **goals**:

- Lean to use the most important features of the MATLAB programming environment
- Understand the differences between analytical and computer-aided (numerical) problem solving
- Learn suitable algorithms for important problems in economics and finance.

This course embraces learning by doing as much as possible, which may result in an above-average workload. There are **7 problem sets** during the course, which count for 50% of the grade. A **programming project** (with presentation), to be solved in groups of two students by the end of the semester, counts for the remaining 50% of the grade. For details on **grading,** see exam section.

After this course, students should be able to reproduce the numerical procedures of current research papers in their field of specialization and to employ numerical methods for their master thesis. This course is organized by methods rather than by applications, however throughout the course examples from economics and finance will be presented.

This course is **specially targeted at MiQE/F, MEcon and MBF** students. All other students are welcome, but may need to do additional work.

Students who whish to own a copy MATLAB can buy a student version in the Rösslitor bookshop at a discounted price of ca. CHF 100. Alternatively, students can work on their projects in the PC Lab 10-U-158, where MATLAB is installed. Students can also use the free programs FreeMat (http://freemat.sourceforge.net/) or Octave (http://www.gnu.org/software/octave/), which are almost identical to MATLAB, albeit less comfortable.

Course structure

The course is organized in **8 blocks of 4 hours each**. Each block is divided into three parts to take students from theory to learning by doing:

- Every block starts with a two-hour **lecture** (8.15-10.00hrs).
- This is followed by a **PC-lab** (ca. 10.15-11.00), where students follow instructions to make their first steps with the new concepts.
- The last hour (ca. 11.15-12.00) is devoted to **supervised self-study**. Students start individual work on the problem sets, but have the opportunity to ask questions to the lecturer. Note that one hour will not always be enough to complete the problem sets. Students are free to work on their own schedule, presence is not required during the supervised self-study.

The course material is divided in eight chapters as listed below, which roughly correspond to the eight lecture units. The program can be adapted to accommodate special requests of the students.

1. Introduction to programming with MATLAB (1)

- Interactive mode
- Variables: scalars, vectors and matrices
- Operators
- Statements

2. Introduction to programming with MATLAB (2)

- Control flow: choice and loops
- Functions (built-in and user defined)
- Graphical output

3. Working with data, linear algebra and econometrics

- Data input and output
- Data manipulation and data cleaning
- Some convenient linear algebra methods: Solving systems of linear equations, eigenvalues and eigenvectors, Choleski decomposition, principal component analysis
- Manually programming the OLS estimator
- An introduction to the Econometrics Toolbox
- Introduction to nonparametric estimation

4. Simulation techniques

- Random number generation
- An introduction to the Monte Carlo method
- Simulating time series processes

5. Nonlinear functions

- Root finding (solving f(x)=0)
- Inverse functions
- Iterative and recursive algorithms
- Approximation and interpolation

6. Optimization

- Metrics
- Convex optimization
- Non-convex and stochastic optimization methods

7. Numerical integration and differentiation

- Numeric integration
- Transform methods (FFT, Fourier-cosine method)
- Numeric differentiation and precision

8. Advanced topics (selection)

- Good programming style
- Finding and avoiding errors
- Accelerating programs
- The symbolic mathematics toolbox
- An introduction to parallel computing
- An introduction to databases

Contextual Studies are considered part of **Contact Learning**; thus, taking part properly implies **regular attendance**. It is the students' own responsibility to ensure that there is **no timetable clash** between the courses they have chosen.

Course literature

- There is a 150-page script for the course.
- All transparencies, sample programs and relevant literature will be made available on the StudyNet.
- Students wishing to prepare for the course can have a look at the "MATLAB Getting Started GUIDE" and some **training videos**, available at the MATHWORKS homepage http://www.mathworks.com/access/helpdesk/help/techdoc/
- The manual for the Econometrics toolbox is available at http://www.spatial-econometrics.com/html/mbook.pdf
- A selection of the vast choice of MATLAB books will be discussed in the first lectures.

Course additional information

Information about the Examination

Examination type

decentral - examination paper written at home (individually) (50%)

Comment: Seven problem sets, to be solved individually.

Examination aids

no regulation necessary

No rules for examination aids are required for this examination.

- For written examinations at home (term paper), courses without credits, etc., no specific rules for examination aids are required.
- The regulations of the University of St. Gallen and the rules of academic work (sources and aids must always be identified) are applicable in a subsidiary fashion.
- All written work must be accompanied by a declaration of authorship.

Question language: **English**

Answer language: **English**

decentral - Group examination paper with presentation (50%)

Comment: Programming project with presentation, groups of 2

Examination aids

no regulation necessary

No rules for examination aids are required for this examination.

- For written examinations at home (term paper), courses without credits, etc., no specific rules for examination aids are required.
- The regulations of the University of St. Gallen and the rules of academic work (sources and aids must always be identified) are applicable in a subsidiary fashion.
- All written work must be accompanied by a declaration of authorship.

Question language: **English**

Answer language: **English**

Examination content

The **problem sets** aim at putting the recently studied material into practise. They comprise the writing of one or more small programs plus possibly a short summary of a topic from the script, from a book or a similar source. It is very important that the problem sets are solved **individually**.

The **programming project** is a larger project that aims at applying several of the techniques presented in the course to one topic (e.g. option pricing, asset pricing, portfolio management, monetary policy, growth theory, parameter estimation and so on). Students can choose from a selection of topics or suggest their own projects. The programming project involves writing one program (possibly with a few support functions), a five-page user documentation and presenting the project. They are solved in **groups of two students**. At the presentation, I will ask one or two questions to each member of the student team.

For both parts, the grading is based on the following factors:

- Technical correctness of the program code
- Mathematical/economical correctness of the solution
- Generality of the solution
- Conformity to the rules of good programming style
- Usability

For the programming project, the following additional criteria apply:

- Difficulty of the problem and depth of the solution
- Quality of the presentation
- Quality of the user documentation
- Level of understanding in the presentation and discussion

The user interface and the graphical design will not be graded.

Exam-relevant literature

This course more is about mastering a technique than reproducing facts, therefore any literature can only be indicative. Furthermore, the required literature will depend on the topic chosen for the programming project.

- The **script**, the **slides** and all **sample programs** as published on the StudyNet
- All other literature items available on the StudyNet, except those marked "for further reading".
- Parts of the MATLAB user guides, available at: http://www.mathworks.com/access/helpdesk/help/techdoc/matlab_product_page2.html#printable_pdf
- **MATLAB getting started guide** (version 7.8), sections 1, 2, 3-1 to 3-63, 4, 5, 7
- **Programming fundamentals** (Version 7.8), pages 1-1 to 1-38; 2-1 to 2-50, 3-1 to 3-38, 4-1 to 4-34, 12
- Note that there is some overlap between these two guides.

Attention please:

We would like to point out to you that this fact sheet has absolute priority over other information such as StudyNet, faculty members' personal databases, information provided in lectures, etc.

When will the fact sheets become binding?
Information about courses: from the start of the bidding process on 26 August 2010
Information about decentral examinations: after the 4th semester week on 18 October 2010
Information about central examinations: from the start of the enrolment period for the examinations on 8 November 2010

Please look at the fact sheet once more after these deadlines have expired.