1. **Title of the module**

ECON8860 – EC886 Computational Methods with Matlab

1. **School or partner institution which will be responsible for management of the module**

Economics

1. **The level of the module (Level 4, Level 5, Level 6 or Level 7)**

Level 7

1. **The number of credits and the ECTS value which the module represents**

15 Credits (7.5 ECTS)

1. **Which term(s) the module is to be taught in (or other teaching pattern)**

Autumn Term

1. **Prerequisite and co-requisite modules**

None

1. **The programmes of study to which the module contributes**

It will be an option module on PhD Economics and PhD Agri-environmental Economics

1. **The intended subject specific learning outcomes.
On successfully completing the module students will be able to:**
	1. be able to understand the theoretical constraints facing computation, including the types of problems that cannot be solved by computers.
	2. be able to understand the impact of these constraints on the design of algorithms.
	3. be able to understand the effect of programming practice on the efficiency of the computation of an algorithm
	4. be familiar with the Matlab environment, including the statistics and optimisation toolboxes, as well as Dynare.
	5. Be able to understand and run basic solution methods in Matlab for economic problems such as monte-carlo analysis, constrained optimisation and value function iteration.
	6. be able to run parallel or distributed jobs on Matlab.
2. **The intended generic learning outcomes.
On successfully completing the module students will be able to:**
	1. self-manage
	2. work with others
	3. work with data
	4. address problems with quantitative analysis
	5. think critically and analytically
	6. use IT and software requiring the use of batch files and command line interfaces
3. **A synopsis of the curriculum**

The aim of the module is to train students to carry out numerical analysis of economic problems using the Matlab software package. The module is intended to be taken up by PhD students at an early stage of their degree with the aim of providing them with core knowledge of the software package (Matlab) and main methodologies that they will use during their degree and into their career. It is also intended to provide a solid foundation for subsequent PhD modules. Because of this, the module will combine both a theoretical component providing an introduction to the core concepts of computation and a practical component, using practical cases and examples carried out in terminal sessions.

Specific topics to be covered include:

* A brief history of computation.
* Introduction to the Matlab package.
* Good programming practice with Matlab
* Using toolboxes
* Parallel/distributed computing
1. **Reading list (Indicative list, current at time of publication. Reading lists will be published annually)**

Attaway, S. (2012) Matlab: a practical introduction to programming and problem solving, 2nd ed., Elsevier.

Brandimarte, P. (2006) Numerical methods in finance and economics: a MATLAB-based introduction

Judd, K. (1998) Numerical Methods in Economics, MIT Press.

Scott, J.C. (2009) But How Do It Know? - The Basic Principles of Computers for Everyone.

1. **Learning and teaching methods**

Total contact hours: 18

Private study hours: 132

Total study hours: 150

1. **Assessment methods**
	1. Main assessment methods

*Weekly Problems Sets (5 at 20% each) (100%)*

13.2 Reassessment methods

Reassessment Instrument: 100% coursework

1. ***Map of module learning outcomes (sections 8 & 9) to learning and teaching methods (section12) and methods of assessment (section 13)***

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Module learning outcome** | *8.1* | *8.2* | *8.3* | *8.4* | *8.5* | *8.6* | *9.1* | *9.2* | *9.3* | *9.4* | *9.5* | *9.6* | *9.7* |
| **Learning/ teaching method** |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Private Study | x | x | x | x | x | x | x | x | x | x | x | x |  |
| Lectures | x | x | x | x | x | x | x | x | x | x | x | x |  |
| **Assessment method** |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Problem Sets | x | x | x | x | x | x | x | x | x | x | x | x |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |

1. **Inclusive module design**

The School recognises and has embedded the expectations of current equality legislation, by ensuring that the module is as accessible as possible by design. Additional alternative arrangements for students with Inclusive Learning Plans (ILPs)/declared disabilities will be made on an individual basis, in consultation with the relevant policies and support services.

The inclusive practices in the guidance (see Annex B Appendix A) have been considered in order to support all students in the following areas:

a) Accessible resources and curriculum

b) Learning, teaching and assessment methods

1. **Campus(es) or centre(s) where module will be delivered**

Canterbury

1. **Internationalisation**

The subject content of the module provides practical tools necessary for mastering computational methods and their application to diverse real-world situations. Acquired knowledge will be necessary to analyse real-world situations in both a national and international context (see module synopsis).

**FACULTIES SUPPORT OFFICE USE ONLY**

**Revision record – all revisions must be recorded in the grid and full details of the change retained in the appropriate committee records.**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Date approved | Major/minor revision | Start date of the delivery of revised version | Section revised | Impacts PLOs (Q6&7 cover sheet) |
|  |  |  |  |  |
|  |  |  |  |  |

Revised FSO Jan 2018