

Course handbook for Part 1

Machine Learning in Finance for Python (ECON5130)

Richard Foltyn

University of Glasgow

13th November 2022

Course information

Course coordinator: Dr. Ankush Agarwal

- Semester 1: Introduction to Python programming
Dr. Richard Foltyn (richard.foltyn@glasgow.ac.uk)
- Semester 2: Introduction to machine learning
Dr. Ankush Agarwal (ankush.agarwal@glasgow.ac.uk)

Intended Learning Outcomes (ILOs)

- 1 Analyse and formulate objective of ML problem and implement solutions using ML libraries and packages in Python.
- 2 Apply different optimisation algorithms and methods through appropriate ML applications.
- 3 Critically analyse different financial datasets while keeping in mind the computational requirements and performance limitations of different ML algorithms.
- 4 Work collaboratively in a group to produce a combined piece of coursework, by liaising with other class members, allocating tasks and coordinating group meetings.

In part 1, we focus on building up knowledge of the Python programming language and libraries required to master ML techniques covered in part 2.

Course structure for part 1

Synchronous activities

- Live lectures in weeks 6–10
 - Tuesdays, 9:00–11:00
 - Lectures consist of hands-on programming to illustrate Python language concepts and how these can be applied to machine learning tasks
 - Students are encouraged to bring laptops to program along
- Computer labs in weeks 7, 9 and 10
 - Wednesdays, 9:00–10:30
 - Students implement exercises while being supported by the teaching team

Asynchronous activities

- Most units contain optional exercises with solutions which students can work through individually

Week 6 (starting October 24)

- Lecture on Tuesday, October 25, 9:00–11:00

Unit 1: **Language and NumPy basics** [[PDF](#)]

- 1 Basic syntax
- 2 Built-in data types
- 3 NumPy arrays
- 4 Optional exercises with provided solutions (asynchronous)

Unit 2: **Control flow and list comprehensions** [[PDF](#)]

- 1 Conditional execution
- 2 Loops
- 3 List comprehensions
- 4 Optional exercises with provided solutions (asynchronous)

Unit 3: **Reusing code – Functions, modules and packages** [[PDF](#)]

- 1 Functions
- 2 Modules and packages
- 3 Optional exercises with provided solutions (asynchronous)

Week 7 (starting October 31)

- Lecture on Tuesday, November 1, 9:00–11:00

Unit 4: **Plotting** [[PDF](#)]

- 1 Line and scatter plots, categorical data
- 2 Labels and annotations
- 3 Multiple plots
- 4 Optional exercises with provided solutions (asynchronous)

Unit 5: **Advanced NumPy** [[PDF](#)]

- 1 Creating and reshaping arrays
- 2 Advanced indexing
- 3 Numerical operations
- 4 Optional exercises with provided solutions (asynchronous)

- Computer lab on Wednesday, November 2, 9:00–10:30
 - Implement first individual exercise (not graded). Submit by Thursday, November 3, 12:00

Week 8 (starting November 7)

- Lecture on Tuesday, November 8, 9:00–11:00

Unit 6: **Handling data with pandas** [[PDF](#)]

- 1 Creating and viewing DataFrames
- 2 Indexing
- 3 Aggregation and reduction operations
- 4 Working with time series data
- 5 Visualisation
- 6 Optional exercises with provided solutions (asynchronous)

Unit 7: **Data input and output** [[PDF](#)]

- 1 I/O with NumPy
- 2 I/O with pandas
- 3 Retrieving macroeconomic / financial data from the web

Week 9 (starting November 14)

- Lecture on Tuesday, November 15, 9:00–11:00

Unit 8: **Random number generation and statistics** [[PDF](#)]

- 1 NumPy's RNG routines
- 2 Statistics functions in SciPy
- 3 Optional exercises with provided solutions (asynchronous)

Unit 9: **Introduction to unsupervised learning** [[PDF](#)]

- 1 Principal component analysis (PCA)
- 2 Introduction to scikit-learn

- Computer lab on Wednesday, November 16, 9:00–10:30
 - Implement second individual exercise. Submit by Thursday, November 17, 12:00

Week 10 (starting November 21)

- Lecture on Tuesday, November 22, 9:00–11:00

Unit 10: **Introduction to supervised learning** [[PDF](#)]

- 1 Linear regression models
- 2 Ridge regression
- 3 Lasso
- 4 Hyperparameter tuning

- Computer lab on Wednesday, November 23, 9:00–10:30
 - Implement third individual exercise. Submit by Thursday, November 24, 12:00

Week 11 (starting November 28)

- Computer lab on Wednesday, November 30, 9:00–10:30
 - Work on group project. Submit by Friday, December 2, 12:00

Both parts 1 & 2 of the course count towards 50% of the final grade.

	Weight
Semester 1	
Week 9: Individual computer exercise	12.5%
Week 10: Individual computer exercise	12.5%
Week 11: Group coursework	25%
Semester 2	
Individual computer exercise	12.5%
Individual computer exercise	12.5%
Group coursework	25%
Total	100%

Individual Computer Exercises

Submission

- Students must submit 2 hand-ins for computer exercises assigned during Wednesday computer labs (submission in week 7 is optional but encouraged)
- Deadline: Thursdays following computer labs (weeks 7, 9 & 10) at 12:00
- Each student submits an individual solution
- All submissions must be done via Moodle
- Submissions should be done in one of two ways:
 - 1 **[Preferred]** Upload your code as a (single) Jupyter notebook (extension `*.ipynb`). Additionally, convert your notebook to PDF using your browser's print function and attach the PDF (a guide on how to create a PDF is included in the [setup document](#) on Moodle).
 - 2 Upload your code as a (single) plain Python file (extension `*.py`). In addition, you need to include any graphs you generated as PDF files as part of your submission.

Assessment

- First submission in week 7 is not graded (formative assessment)
- Submissions 2 & 3 each count towards 12.5% of final grade

Feedback

- Suggested solutions will be posted after deadline

Group coursework

- Programming project to be completed in groups of up to 3 students
- Groups are allocated by course administrators
- Deadline: Friday, December 2, 12:00
- All submissions must be done via Moodle
- Submission format is the same as for the exercises (see above)

Course material & academic support

Course material

- All code can be downloaded from GitHub repository
<https://github.com/richardfoltyn/MLFP-ECON5130>
- Interactive notebooks can be launched directly in the browser (see setup guide on Moodle)
- Moodle additionally contains non-interactive PDF versions of Jupyter notebooks

Academic support

- Please ask Python programming questions in the Python programming forum on Moodle
(<https://moodle.gla.ac.uk/mod/forum/view.php?id=3187200>)
- Office hours for Richard Foltyn (Room 207D, main building)
 - Tuesdays after lectures, 11:00–12:00
 - Wednesdays after labs, 10:30–12:00