**THE UNIVERSITY OF MANCHESTER**

**ALLIANCE MANCHESTER BUSINESS SCHOOL**

### Academic Year 2023/24

**Semester** Semester 1

### Course Unit Code BMAN73701

**Course Unit Title** Programming in Python for Business Analytics

**Credit Rating** 15 credits

### Year PGT course unit

**Course Coordinator and contact details** Dr Fanlin Meng (FM)

Office: 3.094, AMBS

Virtual office: <https://zoom.us/j/94473763354>

[fanlin.meng@manchester.ac.uk](mailto:fanlin.meng@manchester.ac.uk)

Office hours: 1200-1300 Monday

Dr Xian Yang (XY)

Office: 3.077, AMBS

[xian.yang@manchester.ac.uk](mailto:xian.yang@manchester.ac.uk)

Office hours: 1200-1300 Monday

**Other Staff Involved and contact details Qing Yin,** Graduate Teaching Assistant

[qing.yin-2@postgrad.manchester.ac.uk](mailto:qing.yin-2@postgrad.manchester.ac.uk)

### Programme Restrictions This is an elective for MSc Business Analytics and MSc Data Science students

### Pre-requisites None

### Co-requisites Practical experience in programming of any language would be advantageous but not a must

**Dependent course units** None

**Aims**

The aim of this course is to introduce students to the fundamentals of Python, a general-purpose programming language widely used in the application of Data Science, Big Data Analytics and Optimization to business problems. The course will provide the skills for implementing your own algorithms as well as using the thousands of Python packages available for data analysis, modelling, inference, simulation, prediction, forecasting, visualisation, optimization and decision support. The lab classes will provide ample opportunity for students to practice their programming skills and obtain formative feedback. The course is focused on practical knowledge, examples and business applications for data analytics, rather than learning general programming concepts only. The course is very much hands-on with the ultimate goal of turning you into a versatile data analyst for business applications.

The course has an element of group work in the overall assessment (the coursework). The aim of the group work will be to apply the methods and programming skills obtained in the course to a real-world problem and then present your findings as a group.

**Learning Outcomes**

At the end of the course unit, students should be able to:

* Read and write Python code and understand how to use Python packages.
* Implement algorithms of moderate complexity in Python.
* Understand the fundamentals of object-oriented programming using Python.
* Understand how to implement simple data science and optimisation algorithms from the literature to tackle business applications.
* Develop their own algorithms to solve basic data science and optimization problems.
* Use Python packages to solve complex data science, visualisation and optimisation problems in business and management (e.g., portfolio optimization, customer segmentation, and analysis of financial data).

The learning outcomes of the group work (coursework) will be improved skills in working as a group, and improved communication and management report writing skills. These are practical skills that are transferable to team-focused work in general.

**Employability**

This module is designed to teach students key programming and data analytics skills. Although the focus is on programming in Python, one of the most popular programming languages used in industry, the skills around structuring code and problem solving are transferable to other programming languages. The transferability of these skills into the workplace is facilitated by dedicated weekly lab sessions and time in lectures to practice programming skills. The practical group-based coursework will allow students to obtain critical team building and working skills and give students the opportunity to write a management report as a group. These are crucial skills for employability regardless of the industry sector. Formative feedback during lab sessions and on the lecture material will help guiding the students in the correct direction without sacrificing independence.

**Social Responsibility**

This module is designed to teach student how to program responsibly in the sense that your code can be picked by others, understood and used/extended. We will cover topics on commenting code, making use modular, and reusable.

**Equality, diversity and inclusion (EDI)**

The module will promote an inclusive and unbiased classroom culture by encouraging people to participate actively, work in groups during the lecture and the coursework, and also present material in different forms to account for different learning styles, e.g. classroom exercises, online quizzes, youtube videos, supported hands-on exercises during lab sessions.

### Methods of Delivery

The course will be delivered as a block in the first 5 weeks of Semester 1. Each of the 5 weeks will consist of 2 two-hour lectures and 1 two-hour lab session, where the knowledge obtained in the lectures is converted into practical experience; we will also arrange a revision session prior to the exam. The lectures and lab sessions are all face-to-face. Note, due to timetable clashes, there are two teaching slots, one for the MSc Data Science program (Tuesday 4-6pm and Wednesday 1-3pm), and one for the remaining students (Wednesday 10-12 and Friday 1-3pm). Material covered in the lectures and lab sessions will be made available on Blackboard the week before the lecture/lab session. Students will receive formative feedback from the teaching staff and peers on your understanding and application of the taught material. The module will have a GTA (Tao Wen); the GTA and the lecturer will be interacting with students in class and via Blackboard. Quizzes, supporting material, and short videos of the main concepts learnt are provided for each week on Blackboard. Formative feedback is also available for the lecture sessions and made available during the sessions as well as before and after primarily through discussion forums on Blackboard.

It is paramount that students look at the provided material prior to the lectures/labs to avoid getting lost during the delivery as well as make learning as efficient as possible by asking questions on topics requiring clarity. Learning a programming language and being able to apply it to tackle business analytics problems is like learning and using an actual new language. The only way this can be achieved is by sufficient practice. You have 5 intense weeks – with each lecture and lab session build on the material taught the previous weeks – to get up to speed with Python, but it will be worth it as it is a well sought skill on the job market (convince yourself!) that you must mention on your CV (once you obtain it).

***Lecture Hours*** Lectures: 2x2 hours per week x 5 weeks = 20 hours

+ 2 hour revision session prior to exam

***Seminar/Tutorial/Workshop/Lab Hours*** Lab sessions: 2 hours per week x 5 weeks = 10 hours

**Private Study** 118 hours

**Total Study Hours** 150 hours

### Attendance Attendance at all classes is compulsory and will be monitored

### Syllabus and Teaching Schedule

The table below highlights the topics covered in each week of the semester and the lecturer responsible for each week. The lecture will focus more on the theoretical side of the topic, with in-class quizzes and formative feedback provided to support learning, while the lab session focuses on converting the theory into practice. For each week, there will also be made available on Blackboard quizzes and short clips for self-learning and recap of material learnt.

|  |  |  |
| --- | --- | --- |
| **Week** | **Lecturer** | **Topics** |
| **1** | FM | Introduction and getting started with Python |
| **1** | FM | Conditionals and loops |
| **2** | FM | Function, modules and exceptions |
| **2** | FM | Object-oriented Programming |
| **3** | FM | Shallow vs deep copy, reading of and writing to files, variable number of arguments in functions |
| **3** | XY | Numerical analysis |
| **4** | XY | Data Exploration and visualisation |
| **4** | XY | Data processing and preparation in Python |
| **5** | XY | Introduction to machine learning with Python I |
| **5** | XY | Introduction to machine learning with Python II |
| **5** | FM, XY | Revision Lecture |

### Reading List

*Core texts:*

Python manual - <https://www.python.org/doc/>

A.B. Downey. Think Python: How to Think Like a Computer Scientist. O’Reilly, Media, Inc., 2012.

W. McKinney. Python for data analysis: Data wrangling with Pandas, NumPy, and IPython. O'Reilly Media, Inc., 2012.

S. Guido, A. Müller. *Introduction to Machine Learning with Python*: *A Guide for Data Scientists*. O'Reilly Media, 2016.

The course draws material from various sources but these three sources provide a nice overview of all the topics covered in the module.

*Supplementary Texts:*

E. Jones, E. Oliphant, P. Peterson, et al. SciPy: Open Source Scientific Tools for Python. http://www.scipy.org/, 2001-.

C.H. Papadimitriou and K. Steiglitz. Combinatorial optimization: algorithms and complexity. Courier Corporation, 1982.

C. Reeves and J.E. Rowe. Genetic Algorithms: Principles and Perspectives – A Guide to GA Theory. Kluwer Academic Publishers, 2003.

### Assessment

The course has two pieces of assessed work, first a group-based coursework worth 70%, and then two online-tests contributing each 15% (30% in total).

*Group Assignment:*

The first assessment is a group assignment and constitutes in total 70% of your overall grade. For the coursework you will first develop and validate a program that solves a real-world analytics problem (the Python script will be submitted for review), and then, as a group, submit a management report (35%) of around 5000 words and a group presentation (25%). There is also peer assessment to the coursework (10%) to allow group members to award other member of the group some bonus or negative points relative to the group mark. The purpose of the peer assessment is to encourage teamwork and avoid a situation where only a subset of the group members is doing the lion share of the work.

Coursework groups will be allocated in Week 3 after students have finalized their Semester 1 course selections, and the briefing session for the coursework will be in Week 4.

The submission deadline for the executable Python script and the group report, as well as the recorded group presentation, is **3pm on Friday of Week 11**. All submissions are to be made through Blackboard/Turnitin. More information will be made available on the Blackboard coursework section. Any questions arising by students related to the coursework will be put on Blackboard for all students to read.

*Online tests:*

The second assessment are two online tests, each contributing 15% to the overall grade. The online tests will be computer-based, each consisting of around 15 questions and should take between 1 and 1.5 hours to do. The tests are in **Week 7** and **Week 10** and will be done in the labs at AMBS (on a Friday when we had the lab sessions in the first part of Semester 1).

**Resits**

A resit in this course will be assessed with an online exam only.

**Marking Process**

We will follow a fair, rigorous and transparent marking process for all summative work and make available the marking criteria. This should help understand the expectations.

**Feedback**

Feedback in form of written comments for the coursework will be provided via Turnitin, while generic feedback on the online test results will be posted on Blackboard once the test marks have been released.

You can expect feedback for the group assignment and online tests within 15 working days of the submission deadline/online test.

In addition to summative feedback through the coursework and online tests, students will receive also formative feedback during lab sessions and via discussion boards also for the lectures, and through weekly self-learning quizzes available on Blackboard. Each lecture covers also in-class quizzes.

**Methods of Feedback from Students/Course Unit Survey**

A formal questionnaire will be employed at the end of the course unit and used to facilitate continuous improvement in content and delivery of the material.