UNIVERSITY OF LONDON

Course information 2026-27

ST2195 Programming for Data Science

General information

MODULE LEVEL: 5

CREDIT: 30

NOTIONAL STUDY TIME: 300 hours

MODE: Locally Taught, Independent Learner Route and Online Taught

Summary

In the last decade, the demand for programming skills for managing and visualising data has grown remarkably. Python, R, and database management are consistently among the top skills listed in data science and data analyst jobs. Knowing how to write efficient software code to handle and visualise data is essential for any modern data scientist. This course will cover the main principles of computer programming, focusing on data science applications by following the entire pathway from raw data to databases, data wrangling and visualisation, machine learning frameworks, and software development.

Conditions

Please refer to the relevant programme structure in the EMFSS Programme Regulations to check:

- where this course can be placed on your degree structure; and
- details of prerequisites and corequisites for this course.

You should also refer to the Exclusions list in the EMFSS Programme Regulations to check if any exclusions apply for this course.

Aims and objectives

- Gain knowledge of the main principles of programming in the context of data science.
- Develop the ability to handle and visualise data.
- Apply computational thinking in various application domains.
- Provide training in state-of-the-art tools, e.g. SQL, Python, R and Git.
- Communicate the data analysis results to stakeholders and share work with people in the Data Science industry (also using state-of-the-art publishing systems like Quarto).

Learning outcomes

At the end of this half course and having completed the essential reading and activities, students should be able to:

- Convert raw data to relational databases such as SQL.
- Import data to Python and R, apply data manipulation and visualisation.

- Program in Python and R.
- Develop software using best practices for documentation, version control, etc

Employability skills

Below are the three most relevant employability skills that students acquire by undertaking this course which can be conveyed to future prospective employers:

1. Digital skills
2. Complex problem solving
3. Creativity and innovation

Essential reading

McKinney W. Python for Data Analysis, 2nd edition O'Reilly (2017)

Gutagg J.V. Introduction to Computation and Programming using Python, MIT Press, 2nd edition (2017)

Wickham H. and Grolemund G. R for Data Science, 1st edition O'Reilly (2017)

Wickham H. Advanced R., 1st edition Chapman & Hall (2015)

Rammakrishnan R. and Gehrke J. Database Management Systems, 3rd edition, McGraw Hill (2002)

Assessment

This course is assessed by an individual case study piece of coursework (50%) and a three-hour and fifteen-minute closed-book written examination (50%).

Syllabus

Introduction

Data science as an umbrella field with cornerstones data, programming and communication; installing and interacting with R and Python and packages/modules; source code editors and integrated development environments; introduction to the principles of version control, with particular focus on using distributed version control through git, and repository hosting services and collaboration platforms (e.g. GitHub), both directly and through popular IDEs.

Data

Real examples of raw data; definitions and examples of structured, semi-structured, and unstructured data; interacting with popular human-readable file formats for information exchange (e.g. plain text, CSV, XML, JSON, etc.) and others, e.g. spreadsheets and spatial data formats; data types and data structures in R and Python; importing/exporting and interacting with data through core methods and packages.

Relational databases

Relational and non-relational databases; core concepts and terminology in relational database management systems; introduction to SQL and SQLite; creating and manipulating databases; basic SQL queries; creating and manipulating databases from R and Python.

Programming concepts

Core programming concepts and structures (variables, sequences, branches, iterations, control flow structures, objects, classes, functions, scoping); condition and exception handling; debugging techniques.

Data wrangling

Data cleaning and transformation, data representation using tabular data structures and their manipulation; programming and handling data types in R and Python, such as scalars, factors, vectors, matrices, arrays, lists and data frames; introduction to NumPy and Pandas in Python, as well as the data wrangling utilities in base R and the tidyverse collection of R packages.

Exploratory data analysis and visualization

Methods for explanatory data analysis, using various statistical plots such as histograms and boxplots, data visualisation plots for time series data, multivariate data, dimensionality reduction methods for visualisation of high-dimensional data, graph data visualisation methods. Hands-on experience with Python (matplotlib, seaborn) and R (base R graphics, ggplot2).

Machine learning frameworks

Introduction to machine learning via standard frameworks in Python (SciPy, Scikit Learn) and R (glm methods, mlr3, caret); relevant programming concepts such as modularisation and aspects of parallel computing.

Introduction to software development

Software development phases and life cycle methodologies; development of R packages; development of Python packages and modules; publishing packages and models; documenting R and Python code; test-driven development in R and Python; interaction with large language models to support software development.

Publishing systems

Create dynamic content in R and Python using established publishing systems (e.g. RMarkdown and Jupyter notebooks) and modern ones (Quarto); publish reproducible, production-quality articles, presentations, dashboards, websites, blogs, and books in various formats (e.g. HTML, PDF, MS Word, PowerPoint, etc); introduction to Shiny.