DATA ENGINEERING FOR THE SOCIAL WORLD (ME204)

Course Duration: 54 hours of lecture and class time (Over three weeks)

Summer School Programme Area: Research Methods, Data Science, and Mathematics

LSE Teaching Department: Data Science Institute

Lead Faculty: Dr Jonathan Cardoso-Silva (Data Science Institute)

Pre-requisites: Students should already be familiar with computer programming at an introductory level (variables, if-else, loops, functions). In the past, we have welcomed complete beginners to this course, and many have done well, but it can be a tough learning curve! If you'd like to prepare in advance, we recommend focusing on Python basics. A great starting resource is Chapters 1-5 of the book *Automate the Boring Stuff with Python* by Al Sweigart, which is freely available online.

Course Description:

Data science has unlocked exciting possibilities for social scientists through its diverse toolkit, including big data analysis, visualisation, and machine learning models, enabling them to extract valuable insights from their data. Yet, the success of a data-driven project hinges on data quality. This is where data engineering plays a pivotal role. Professionals must ensure that their acquired data is sufficient and accurate and must be adaptable to handle 'messy data' effectively.

A substantial portion of time in data-driven projects (anecdotally 80%) is dedicated to cleaning and preprocessing data, with only 20% said to be devoted to building, evaluating, and deploying machine learning models. Despite the emergence of new AI technologies, which promise to automate many coding tasks, data manipulation is likely to remain an indispensable skill due to the inherent messiness of real-world data.

In this course, you will learn the fundamentals of data engineering, including:

- Reasoning about the structure and format of data
- Collecting data from real websites and APIs
- Best practices for efficient data storage
- Basics of the SQL language
- Tools available in the programming language Python for data pre-processing and reshaping
- Using Generative AI tools (ChatGPT and GitHub Copilot) to write and debug code efficiently
- Organizing data into a 'tidy' format, suitable for future analysis
- Conducting exploratory data analysis, including static and dynamic visualisations.
- Building simple websites to report and communicate your findings effectively

By the end of this course, you will be proficient in producing a website to communicate your collected data and showcase your newly acquired data-wrangling abilities.

Reading:

The main reading material will consist of ***lecture slides and related materials*** distributed at the beginning of the course. Optional further reading is also recommended from the following textbooks

- Sweigart, Al. *Automate the Boring Stuff with Python*. 2nd edition. No Starch Press, 2019.
- VanderPlas, Jake. *Python Data Science Handbook: Essential Tools for Working with Data* 2nd edition. Sebastopol, CA: O'Reilly Media, Inc, 2022.
- Lutz, Mark. *Learning Python*. 5th edition. Sebastopol, CA: O'Reilly Media, 2013.
- Tanimura, Cathy. *SQL for Data Analysis: Advanced Techniques for Transforming Data into Insights*. First edition. Sebastopol, CA: O'Reilly Media, 2021.
- Wilke, C. Fundamentals of Data Visualization: A Primer on Making Informative and Compelling Figures First edition. Sebastopol, CA: O'Reilly Media, 2019.

Course Structure:

- Lectures: 36 hours
- Classes: 18 hours

Assessments:

Formative assessments: There will be daily lab exercises which will not contribute to the final grade but will allow students to test their understanding of the content and prepare them for

Problem set: A mid-term problem set. It will count for 25% of the overall grade.

Final project: Students must submit a dashboard displaying data collected during the course and showcase their newly acquired data-wrangling abilities. The project will count for 75% of the overall grade. The precise timing of submission of the final project will be communicated during the course.

Lecture Schedule:

WEEK 01

1. The Data Science Toolbox (Mon 14 Jul)

Course overview and introduction to the data science toolbox:

- What is this course all about?
- What do we mean by data science and by data engineering?
- Why Python is a fantastic tool for data manipulation
- The fundamentals of the Python language and a look into the NumPy and pandas libraries

- Documenting analysis effectively with Markdown in Jupyter Notebooks
- Practice: Explore and manipulate sample datasets using these libraries

2. Data Types and Common File Formats (Tue 15 Jul)

- Distinguishing structured vs unstructured vs semi-structured data
- Primitive types, objects, and data frames
- Overview of standard file formats: CSV, JSON and XML
- Practice: Load and inspect data from these formats using Python

3. Data Visualisation Basics (Wed 16 Jul)

- Introduction to data visualisation principles
- Overview of the Matplotlib library: creating static plots
- Introduction to Seaborn: enhancing visualisations with themes and statistical plots
- How to effectively communicate your insights: guidelines on clear titles, avoiding overly complicated charts, where to use rich and dense visualisations.
- Using Generative AI to do the 'boring bits' of dataviz code for us
- Practice: Creating line plots, scatter plots, and bar charts using Python

4. Collecting Data from APIs: OpenMeteo (Thu 17 Jul)

- Introduction to APIs and their importance
- Using Python's requests library to interact with APIs
- Querying the OpenMeteo API for weather data
- Practice: Fetching, parsing, and saving API data into files

WEEK 02

5. Mastering JSON: From Chaos to Clarity (Mon 21 Jul)

- Why is JSON often nested, and how does normalization simplify working with it?
- Techniques for flattening nested JSON structures
- Handling inconsistent JSON schemas in real-world data
- Write clear prompts to simplify JSON normalization using AI tools
- Best practices for JSON handling in pandas

6. From Loops to Vectorization (Tue 22 Jul)

- The concept of vectorized operations in Python
- Why vectorized code is more efficient than loops
- Hands-on: Converting pure Python loops into pandas operations with the help of Generative AI tools

- Practice: Optimize a real-world data transformation task using pandas

7. Working with Authenticated APIs (Wed 23 Jul)

- Overview of API authentication methods (e.g., API keys, OAuth)
- Hands-on: Using an API that requires authentication (e.g., Spotify API, Reddit API or YouTube API)
- Parsing nested JSON data from authenticated requests
- Practice: Cleaning and exploring fetched API data

8. Getting Started with Git and GitHub (Thu 24 Jul)

- Introduction to version control with Git
- Basics of committing, pushing, and branching
- Using GitHub Pages to create simple websites from Markdown files
- Practice: Create a repository, make commits, and manage a simple collaborative project

WEEK 03

9. Intro to Databases (Mon 28 Jul)

- Fundamentals of databases: tables, primary and foreign keys
- When to use databases over files
- Basic SQL queries (SELECT, GROUP BY, ARRANGE, etc.)
- Practice: Querying SQLite databases using Python

10. Data Merging and Integration(Tue 29 Jul)

- Joining data from different sources
- Exploring scenarios for combining datasets with real-world examples
- Merges and joins in pandas: inner, outer, left, and right joins
- Practice: Merging multiple datasets in Python and SQL and troubleshooting mismatched data

11. A Primer on Web Scraping (Wed 30 Jul)

- How does web scraping differ from working with APIs, and when should you choose one over the other?
- Ethics and legal considerations of web scraping
- Using Python's scrapy library to parse HTML content
- Extracting data from simple web pages and handling HTML structures
- Practice: Scraping and cleaning data from a public website

12. Managing your data pipeline (Thu 31 Jul)

- Practical advice for using GitHub
- Writing compelling data-driven reports
- Data versioning & reproducibility
- Practice: Setting up a GitHub repo with effective documentation

Content of Computer Labs:

The labs will consist of guided computer exercises

1. Python basics: understanding variables, loops, and functions.
2. Reading and writing data file formats, including CSV and JSON.
3. Practicing data visualization with Matplotlib and Seaborn.
4. Collecting data from APIs using Python's requests library.
5. Cleaning and organizing nested JSON data effectively.
6. Converting Python loops into efficient vectorized pandas operations.
7. Preprocessing and preparing data fetched from APIs for analysis.
8. Setting up GitHub repositories and creating simple websites with Markdown.
9. Designing and using SQLite databases for structured data storage.
10. Combining multiple sources of data using SQL and pandas
11. Supervised project support sessions.
12. Supervised project support sessions.

Credit Transfer: If you are hoping to earn credit by taking this course, it is advisable that you confirm it is eligible for credit transfer well in advance of the start date. Please discuss this directly with your home institution or Study Abroad Advisor.

As a guide, our LSE Summer School courses are typically eligible for three or four credits within the US system and 7.5 ECTS in Europe. Different institutions and countries can, and will, vary. You will receive a digital transcript and a printed certificate following your successful completion of the course in order to make arrangements for transfer of credit.

If you have any queries, please direct them to summer.school@lse.ac.uk