UNIVERSITY OF WARSAW

Course syllabus

Course title	Advanced Python for cognitive scientists
Instructor(s)	Marcin Leśniak PhD
Contact details	marcin.lesniak@psych.uw.edu.pl
Affiliation	Faculty of Psychology, University of Warsaw
Course format	Class, workshop
Number of hours	45 hours
Number of ECTS credits	**4 ECTS credits**
Brief course description	The goal of the course is to build fluency in using Python programming language as a tool for scientific computing, data manipulation and visualization. We will introduce libraries which constitute a core of Python ecosystem for data analysis: numpy, scipy, pandas, matplotlib. After covering the basics, students will have the opportunity to hone their skills by working through a number of applications of the introduced tools in data analysis. Simultaneously, they will be improving their programming style and learning about good programming practices. Previous experience with Python is necessary.
Full course description	This course is designed as a continuation of an introductory course of Python programming. It is assumed that students know the basics of language syntax and are able to write simple programs on their own. In this class they will expand their knowledge of the language, get to know popular Python libraries, and learn practical applications of their skills. In addition to imperative style of programming, already known to students, concepts of high-level array programming (based on numpy and pandas libraries) are introduced. The focus is on scientific computing and exploratory data analysis. Libraries covered include numpy, scipy, pandas, matplotlib. Students learn important aspects of data literacy: data preprocessing, data manipulation, data visualization. These practical skills are prerequisites for delving deeper into issues of computational modeling and data science.
Learning outcomes	Student knows and understands: - main Python libraries for data analysis (K_W04) - concepts of exploratory data analysis and data visualization (K_W08) Student is able to: - perform basic data analysis, build data preprocessing pipeline, program experimental procedure in Python (K_U02, K_U04).

	- search for proper tools and software libraries to solve a particular task, experiment with different approaches (K_K01, K_K02)
Learning activities and teaching methods	The class will be conducted in a computer laboratory and/or in an online environment based on Zoom. It will consist of programming exercises interspersed with short lectures and demonstrations. Homework assignments are to be expected.
List of topics/classes and bibliography	1. Review of basic Python syntax and data types. Jupyter Notebook. 2. Advanced iteration utilities and data structures in Python. 3. Working with strings and text. Working with files ("flat databases"). 4. Foundations of numerical computing with Numpy. 5. Advanced features of Numpy. Vectorization and broadcasting rules. 6. Data frames and Pandas. Indexing, vectorization and label-based broadcasting model. 7. Data processing with Pandas. Aggregation, relational operations and method chaining syntax. 8. Data visualization. Pandas, Numpy and Matplotlib. 9. More on visualization. More on Matplotlib. Statistics in Python. Recommended (not obligatory) readings: 1. Sheppard, K. (2016). Introduction to Python for Econometrics, Statistics and Numerical Analysis: Third Edition https://www.kevinsheppard.com/files/teaching/python/notes/python_introduction_2019.pdf 2. Rougier, N.P. (2017). From Python to Numpy http://www.labri.fr/perso/nrougier/from-python-to-numpy/ 3. McKinney, W. (2017). Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython: Second Edition
Assessment methods and criteria	**In-class quizzes (50%)** These assessments, conducted at the start of each class, will encompass material from the prior class and assigned homework (practical programming tasks), ensuring a holistic evaluation of your understanding. The quizzes will include multiple-choice and short-answer questions. **Final Practical Programming Test (50%):** This assessment will feature a series of hands-on programming tasks conducted during the final class. You will be tasked with applying your acquired programming skills to solve real-world problems, demonstrating your practical proficiency. Grading will be based on the following criteria: 95% or more = 5! 90-94% = 5 80-89% = 4.5 70-79% = 4

	60-69% = 3.5 50-59% = 3 below 50% = 2 (fail)
Attendance rules	Attendance to the workshop is obligatory. Two unexcused absences are allowed in the semester. Further unexcused absences may result in lowering the grade.
Prerequisites	“Introduction to programming in Python” class or equivalent.
Academic honesty	Students must respect the principles of academic integrity. Cheating and plagiarism (including copying work from other students, internet or other sources) are serious violations that are punishable and instructors are required to report all cases to the administration.
Remarks	Group 1: Tuesday 8:30-12:30; Group 2: Thursday 8:30-12:30