NRSC 7657 Workshop in Advanced Programming for Neuroscientists

Course Description

The goal of this course is to provide neuroscience students with practical skills in modern programming to facilitate their research and understand technical development in the field. It is project-based mixed with practical pedagogy in computational tools and techniques for a broad range of topics within neuroscience. The course will cover concepts in programming, data types, workflow, data and code management, and collaboration tools. Students are expected to have completed some very basic introduction to programming prior to this course, or have previous experience with scientific computing in either MATLAB or Python. **Students are expected to complete an independent project using Python or (/and) MATLAB, using data from their own work or publicly available datasets**. The course will use both Python and MATLAB for didactic sessions, with a goal of providing familiarity in both platforms used in neuroscience.

The course is taught over 10 weeks, with one 3 hour session per week. Each session is divided between lecture and active programming of examples or independent coding. Several sessions in the latter half of the course are dedicated to group programming time, and students are expected to attend these sessions and encouraged to work openly during these times. All students will create and maintain a GitHub repository for their final project.

Schedule

~~June 5~~- Aug 14 2023 | Mondays 1 – 4PM | No class Monday July 3
June 12

Week 1 – overview / review	Course overview: theory of computing, landscape of computing options. Basic usage in python and MATLAB; basic data types; environments Style guidelines (ten simple rules); git and version control
Week 2 – language fundamentals	Functions; Objects and Classes; Workspaces Typical data formats: working with tabular data, images, and time series. NeurodataWithoutBorders format
Week 3 – workflow management and outputs	Importing and exporting Plotting and visualization - from bar charts to 3D animation
Week 4 – usability	Troubleshooting and debugging; unit testing
Week 5 – scaling	Iteration and code profiling; parallel computing. Code quality-of-life topics
Week 6 – collaboration	Cloud-based tools: AWS, GCC, Colab, jupyterhub, deepnote. Overview of some available SAAS tools, python focused.
	Group programming time
Week 7 – applications/flex topic	Applications: image processing (ES)
	Group programming time
Week 8 – applications/flex topic	Applications: spike sorting (DD)
	Group programming time
Week 9 – applications/flex topic	Group programming time
	Group programming time / Final pres. and code review (if big class)
Week 10 – Final presentations and code review	Final pres. and code review
	Final pres. and code review

Evaluation

Students will be evaluated on lecture attendance, participation, and effort towards completing their final project. Weekly progress reports on the final coding project are expected to be uploaded to the project GitHub repository.