# Python for Neuroscience Course Report 2023/2024

## Summary Points

- High demand for Python in neuroscience graduate programs.
- The course duration was extended to 32 hours, allowing for more time on fundamentals.
- Frontal lectures with in-class exercises and practicals were effective.
- Inclusion of final projects related to research, submitted via GitHub.

## Timeliness and Need

The course had good participation with 20 sign-ups from students and postdocs, and 12 students completing assignments for credits. The inclusion of Python in the neuroscience curriculum is seen as essential due to the increasing use of Python-based tools in the field. The addition of TAs, who were previous students, significantly improved student engagement and learning.

## Overall Organization

The course aimed to teach Python in a practical, research-oriented manner. Starting with basic concepts, the course was designed to ensure students could apply what they learned directly to their research work. The extended 32-hour schedule was well-received, though some students suggested even more time would be beneficial.

## Structure of the Lectures

The course was structured using Jupyter Notebooks and included: - 20-30 minute blocks of explanations followed by 20-30 minute hands-on exercises. - Office hours for personalized assistance. - TAs providing additional support during practical sessions.

## Lectures Content

The syllabus covered: 1. **Basic Python**: Variables, control flow, functions, classes, and objects. 2. **Scientific Python**: `numpy`, `pandas`, and `matplotlib`. 3. **Real-World Python**: Local installations, environments, `pip`, and basic data analysis. 4. **Special Topics**: Students chose from experiment scripting, functional imaging analysis, advanced statistics, introduction to neural networks, and data visualization.

Based on feedback, more time was dedicated to practical exercises, and complex topics were better titrated to match student needs.

## Evaluation

Students completed assignments after the basic and scientific Python modules and are currently a

Python project relevant to their research. Projects will be submitted via GitHub, encouraging code sharing and version control practices.

## Conclusion

The adjustments made this year, including extended hours, practical focus, and the involvement of TAs, improved the overall effectiveness of the course. For future iterations, I feel that the current balancing works well. The only option to consider would be to offer two separate introductory and advanced courses to cater to different student needs.