A Good Start at Python for Scientists

Not drowning in the ocean of Python ressources - Presentation

The many ways of running Python code

Make sure everything work for everyone

1. Interactive - Demo
    - REPL (Read–eval–print loop) aka Python shell
        - Easiest way to test an installation
        - Gives info about version
    - Ipython terminal
    - VSCode interactive windows
        - ctrl + maj + P for list of command in VSCode
2. Run scripts - Demo
    - Terminal
    - VSCode
    - Part of a script in VSCode
3. Jupyter notebooks - Demo
    - Standard
    - VSCode

Modules, their documentation, and how to use both

1. Installing packages - Demo
    - `pip` and `conda`
    - Try `help` and `list` with both
2. The dot syntax - Demo
    - Submodules
    - Objects
3. Using modules - Demo
    - `import` statement
    - `as` statement
    - `from` keyword
    - Traditions
4. Documentation
    - How to read a function signature - Presentation
    - Access documentation - Demo
        - Websites
        - In Ipython
        - In VSCode

Numpy: think like a vector

Theory - Demo

1. Initialization
    1. `np.array` and `np.asarray`
    2. Zeros and ones
    3. `np.arange` and `np.linspace`
    4. random
    5. Data type `dtype`
    6. `ndarray.shape`
2. Elementwise operations
3. Reduction operations
4. Indexing
    1. Multidimensional indices
    2. Adding dimensions to broadcast
    3. Masking

Exercise

Matplotlib: visualize everything

Theory - Demo

1. The two ways of plotting
    1. Pyplot
    2. Figure and axes with `plt.subplots`
2. `plt.show` and common problems with interactive python
3. Customize your plots
    1. Keyword arguments
    2. Figures and axes through `ax.update`
    3. Global parameter with `matplotlib.rc`
4. `fig.savefig`

Exercise

Scipy: add some science - Demo

1. Passing a function to a function
2. Pay attention to return types

Exercise

Final exercise: A guided tour through a complex example using several libraries