Tutorial on Python Programming for Computational Imaging

Frank Ong

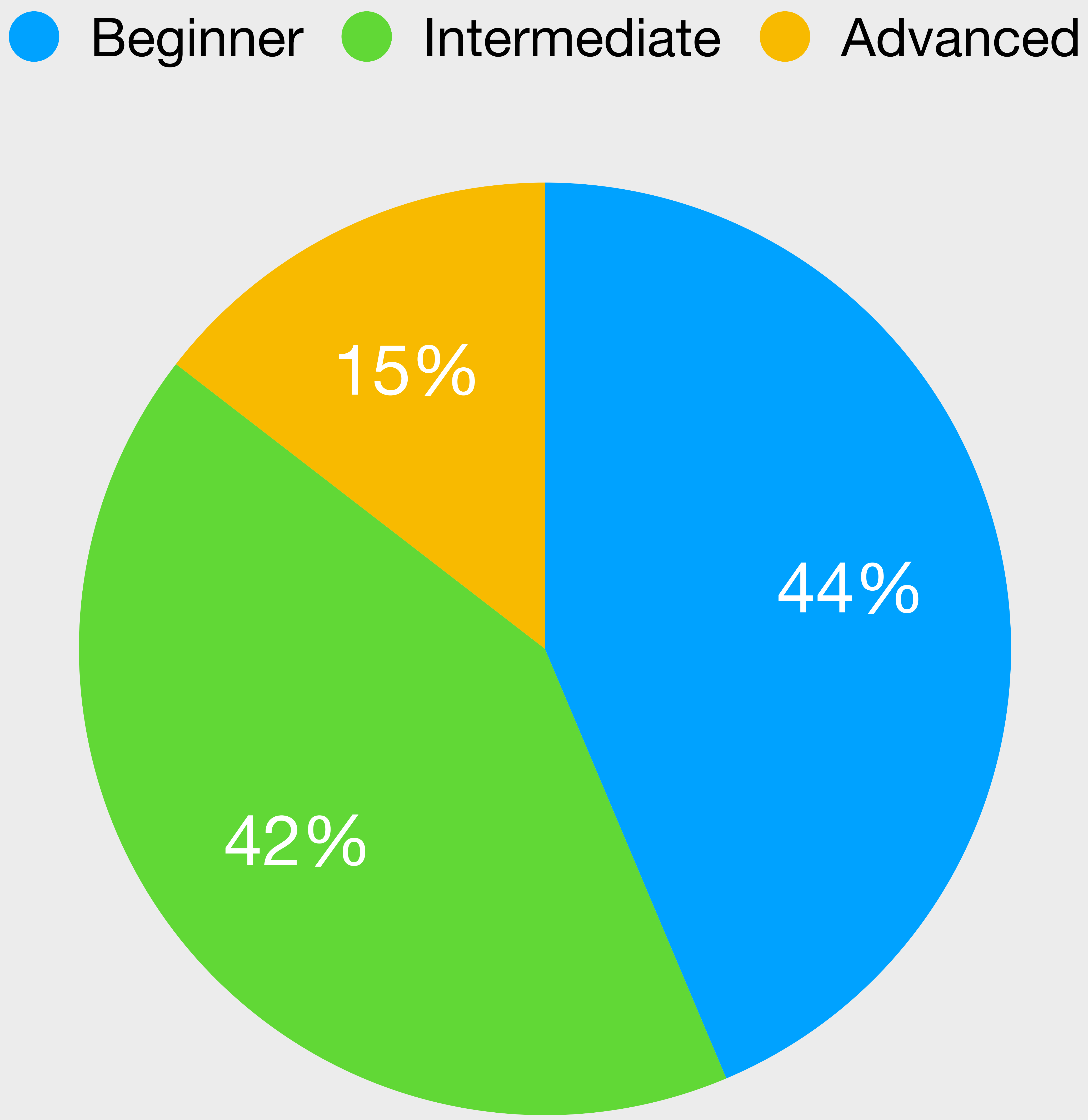
IMA Workshop on Computational Imaging 2019

My Experience with Python

- Work on MRI reconstruction
- Many years: Matlab for prototyping, C for deploying
- Introduced to scientific Python while TAing
  - Blown away by its flexibility and simplicity
- Now: Python for prototyping and deploying

Outline

- Overview of scientific Python (slides)
- Tutorial (notebooks)
    - Basics
    - Fourier compressed sensing



Python

- Interpreted, high-level, general-purpose language
- Started in December 1989 by Guido Van Rossum
  - "A hobby programming project during Christmas"
  - Name from *Monty Python*
- Design philosophy: simplicity, readability, practicality… ("Zen of Python.")

http://python-history.blogspot.com/2009/01/pythons-design-philosophy.html

Scientific Python Packages

- **NumPy (2006):**
  - An efficient multidimensional array object,
  - Fast mathematical operations over arrays
  - Linear Algebra, Fourier Transforms, Random Number Generation

- **Scipy (2001):** high-level numerical routines. Optimization, interpolation, filters…

- **Matplotlib (2003):** 2-D visualization, “publication-ready”plots

- **IPython (2001):** an advanced Python console for interactive computing

- **Jupyter notebook (2011):** web-based interactive comp. for Julia, Python and R

TensorFlow
PYTORCH
Pyfolio
SymPy
NetworkX
SM StatsModels Statistics in Python
scikit-image image processing in python
scikit learn
matplotlib
PyMC3
Bokeh
pandas
$y_{it} = \beta' x_{it} + \mu_i + \epsilon_{it}$
xarray
NumPy
jupyter
SciPy
IP[y]: IPython
Cython
DASK
python™
Numba

(and many, many more)

Python's Strengths (IMO)

- **Batteries included** - rich libraries
- **Universal -** a language used for many different problems
- **Free and open source**
- **Easy to learn** - simple syntax
- **Efficient code -** quick development times and quick execution times

Python's Weaknesses (IMO)

- **Not domain specific, verbose commands:** A.conj().T instead of A'
- **No simple debuggers like Matlab's**
- **No easy solution for shared-memory parallelism**
  - Partly due to design (Global Interpreter Lock)

Tutorial

- Basics using NumPy and Matplotlib
- Fourier compressed sensing

- https://github.com/frankong/ima2019_python_tutorial

launch binder