The Unexpected Effectiveness of Python in Science

Jake VanderPlas @jakevdp
PyCon 2017
PyCon’s Mosaic
$ whoami
jakevdp
Blog: [http://jakevdp.github.io](http://jakevdp.github.io)


Code:

```bash
$ whoami
jakevdp
```
$ whoami
jakevdp

“Wow, I’ve never met an astronomer before.”
Astronomy
Then . . .
Astronomy Now . . .

Hubble Space Telescope

Source: http://spacetelescope.org/

Sloan Digital Sky Survey

Source: http://sdss.org/
Hubble’s “Ultra Deep Field”

Source: http://spacetelescope.org
Hubble’s “Ultra Deep Field”

Source: http://spacetelescope.org
Astronomy in the 21st Century . . .

TRAPPIST-1 Exoplanetary System

TRAPPIST-1 Exoplanetary System
Kepler (K2) Observations

K2 View of TRAPPIST-1
15 Dec 2016 20:55

Kepler Telescope: NASA
K2 Data: Ethan Kruse
Kepler/K2 Guest Observer Office
http://keplerscience.arc.nasa.gov  KeplerGO@mail.arc.nasa.gov

Repositories

PyKE
A suite of Python tools to analyze Kepler/K2 data
kepler  k2
Python  7  9  Updated 8 minutes ago

KeplerScienceWebsite
Website for the Kepler/K2 Science Center.
HTML  6  Updated 16 minutes ago

Top languages
- Python
- CSS
- Makefile
- HTML

People
This organization has no public members. You must be a member to see who’s a part of this organization.
James Webb Space Telescope (JWST)

Source: NASA
James Webb Space Telescope (JWST)

Source: NASA/JWST
James Webb Space Telescope (JWST)

Starlight filters through the planet's sodium-rich atmosphere.

Source: NASA/JWST
Large Synoptic Survey Telescope
8.4-meter Primary Mirror
3 Gigapixel Camera

Focal plane

Utility Trunk—houses support electronics and utilities

Cryostat—contains focal plane & its electronics

1.65 m (5' 5'')

L1 Lens
L2 Lens
L3 Lens
Filter

Camera 3/4 Section
3 Gigapixel Camera

Camera ¾ Section

Utility Trunk—houses support electronics and utilities

Cryostat—contains focal plane & its electronics

L3 Lens

L2 Lens

L1 Lens

Filter

1.65 m (5’-5’’)

18x6 to 837x482
3 Gigapixel Camera

Utility Trunk—houses support electronics and utilities

contains focal its electronics

1.65 m (5'-5'')

L3 Lens

Filter

L2 Lens

L1 Lens

Camera 3/4 Section
3 Gigapixel Camera = ~1500 HD TVs
- Survey mode: 2 exposures every ~30 seconds
- Images the full southern sky every three nights for a decade
- 15-30 TB/night!
- Final 10-year catalog: 100s of Petabytes
What will we do with all this data?

(Left as a 616-page exercise for the reader)

https://www.lsst.org/scientists/scibook
LSST
Large Synoptic Survey Telescope - Astronomy that's Wider, Faster, Deeper
Tucson, AZ  http://lsst.org

Repositories

Repositories

Search repositories...

Type: All

Language: All

afw
LSST data management: astronomical framework, including images and tables

python
c-plus-plus
astronomy

C++  5  8  Updated an hour ago

pipe_tasks
LSST Data Management: astronomical data processing tasks

Top languages

Python  Shell  C++  TeX
Makefile

People

18
Mentions of Software in Astronomy Publications:

Compiled from NASA ADS (code).

Thanks to Juan Nunez-Iglesias, Thomas P. Robitaille, and Chris Beaumont.
The Unexpected Effectiveness of Python in Science
But Why Python?

Python is a “teaching language”

... created to “bridge the gap between the shell and C”

“never intended. ... to be the primary language for programmers.”

Guido Van Rossum The Making of Python
“I thought we'd write small Python programs, maybe 10 lines, maybe 50, maybe 500 lines — that would be a big one”

Guido Van Rossum  The Making of Python
Why is Python such an effective tool in science?
Why is Python such an effective tool in science?

1. Interoperability with Other Languages
“If I have seen further, it is by standing on the shoulders of giants.”

- Isaac Newton
“If I have seen further, it is by importing from the code of giants.”

- Definitely Not Isaac Newton
“Scientists... work with a wide variety of systems ranging from simulation codes, data analysis packages, databases, visualization tools, and home-grown software-each of which presents the user with a different set of interfaces and file formats. As a result, a scientist may spend a considerable amount of time simply trying to get all of these components to work together in some manner...”

- **David Beazley**
  Pythonista Extraordinaire
  *Scientific Computing with Python*
  (ACM vol. 216, 2000)
“I had a hodge-podge of work processes. I would have Perl scripts that called C++ numerical routines that would dump data files, and I would load them up into MatLab to plot them. After a while I got tired of the MatLab dependency... so I started loading them up in GnuPlot.”

- John Hunter
  creator of Matplotlib
  SciPy 2012 Keynote
Science Before Python . . .

“My advisor had a heavily customized awk/sed/bash workflow to manage job submissions and postprocessing of C codes for supercomputing runs... So I used her scripts to run my jobs, and on top of that had added my own layer of Perl, plus a hefty amount of Gnuplot, IDL and Mathematica.”

- Fernando Perez
  creator of IPython
  via email
Python is Glue.
Python glues together this hodge-podge of scientific tools.

High-level syntax wraps low-level C/Fortran libraries, which is (mostly) where the computation happens.

Python is Glue.
Why is Python such an effective tool in science?

1. Interoperability with Other Languages

2. “Batteries Included” + Third-Party Modules
Python has built-in libraries for *nearly* everything . . .

. . . and there are third-party libraries for everything else.
The Genesis of Scientific Python

“Prior to Python, I used Perl (for a year) and then Matlab and shell scripts & Fortran & C/C++ libraries. When I discovered Python, I really liked the language... But, it was very nascent and lacked a lot of libraries. I felt like I could add value to the world by connecting low-level libraries to high-level usage in Python.”

- Travis Oliphant
  creator of NumPy & SciPy
  via email
Python’s Scientific Stack
Python’s Scientific Stack

IPython
NumPy
jupyter
DASK
Python
Numba
Python’s Scientific Stack

- matplotlib
- pandas
- xarray
- NumPy
- Jupyter
- SciPy
- IPython
- Dask
- Numba
- Bokeh
- Cython
- Python
Python's Scientific Stack

- StatsModels
- SymPy
- NetworkX
- scikit-image
- matplotlib
- pandas
- PyMC
- xarray
- NumPy
- jupyter
- SciPy
- IPython
- Dask
- python
- Numba
Python's Scientific Ecosystem

(astropy, biopython, DIPY, NIPY, SunPy, and many, many more)

StatsModels
(Statistics in Python)

SymPy

NetworkX

scikit-image
(image processing in python)

scikit-learn

matplotlib

pandas

PyMC

xarray

NumPy

jupyter

SciPy

IPython

DASK
Why is Python such an effective tool in science?

1. Interoperability with Other Languages
2. “Batteries Included” + Third-Party Modules
3. Simplicity & Dynamic Nature
YOU'RE FLYING! HOW?

I LEARNED IT LAST NIGHT! EVERYTHING IS SO SIMPLE!
HELLO WORLD IS JUST PRINT "Hello, world!"

I DUNNO... DYNAMIC Typing? WHITESPACE?
COME JOIN US! PROGRAMMING IS FUN AGAIN! IT'S A WHOLE NEW WORLD UP HERE!
BUT HOW ARE YOU FLYING?

I JUST TYPED import antigravity
THAT'S IT?
... I ALSO SAMPLED EVERYTHING IN THE MEDICINE CABINET FOR COMPARISON.
BUT I THINK THIS IS THE PYTHON.
Python Enters Science:

“Python is a language that is very powerful for developers, but is also accessible to Astronomers. Getting those two classes of people using the same tools, I think, provides a huge benefit that’s not always noticed or mentioned.”

- Perry Greenfield
  Space Telescope Science Institute
  PyAstro 2015
Often-overlooked fact . . .

For day-to-day scientific data exploration, **speed of development** is primary, and **speed of execution** is often secondary.
Why don't you use C instead of Python? It's so much faster!
Why don't you use C instead of Python? It's so much faster!

Why don't you commute by airplane instead of by car? It's so much faster!
Ada Marie did what scientists do:
She asked a small question,
and then she asked two.
And each of those led her
to three questions more,
And some of those questions resulted in four.
Jupyter notebooks embody this kind of quick, nonlinear exploration:
Why is Python such an effective tool in science?

1. Interoperability with Other Languages
2. “Batteries Included” + Third-Party Modules
3. Simplicity & Dynamic Nature
4. Open ethos well-fit to science
Does social science have a replication crisis?

By Joshua Tucker  March 9
Psychology's Replication Crisis Has a Silver Lining

It's an opportunity for the field to lead.

There is a crisis in psychology. It's not those rare cases of outright fraud, as when the social psychologist Diederik Stapel simply made up the results of dozens of...
A sense of crisis is developing in economics after two Federal Reserve economists came to the alarming conclusion that economics research is usually not replicable.

The economists took 67 empirical papers from 13 reputable academic journals and replicated or reanalyzed the results. Almost half failed to replicate, with a massive 70% of the research being rejected.
Cancer Research Is Broken

There’s a replication crisis in biomedicine—and no one even knows how deep it runs.

By Daniel Engber
Big Science is broken

Pascal-Emmanuel Gobry
The replication crisis in science has just begun. It will be big.

Summary: After a decade of slow growth beneath public view, the replication crisis in science begins breaking into public view. First psychology and biomedical studies, now spreading to many other fields — overturning what we were told is settled science, the foundations of our personal behavior and public policy. Here is an introduction to the conflict (there is pushback).
Big Science is broken

The replication crisis will be big.

Summary: After a decade of scientific breakthroughs begins breaking into public consciousness, many other fields — overtaking personal behavior and public health — are in crisis too.
“An article about computational result is advertising, not scholarship. The actual scholarship is the full software environment, code and data, that produced the result.”

–Buckheit and Donoho (1995)
LIGO Gravitational Wave Event (GW150914)
LIGO Gravitational Wave Event (GW150914)
SIGNAL PROCESSING WITH GW150914 OPEN DATA

Welcome! This ipython notebook (or associated python script GW150914_tutorial.py) will go through some typical signal processing tasks on strain time-series data associated with the LIGO GW150914 data release from the LIGO Open Science Center (LOSC):

- https://losc.ligo.org/events/GW150914/
- View the tutorial as a web page - https://losc.ligo.org/s/events/GW150914/GW150914_tutorial.html/
- Download the tutorial as a python script - https://losc.ligo.org/s/events/GW150914/GW150914_tutorial.py/
- Download the tutorial as iPython Notebook - https://losc.ligo.org/s/events/GW150914/GW150914_tutorial.ipynb/

To begin, download the ipython notebook, readligo.py, and the data files listed below, into a directory / folder, then run it. Or you can run the python script GW150914_tutorial.py. You will need the python packages: numpy, scipy, matplotlib, h5py.

On Windows, or if you prefer, you can use a python development environment such as Anaconda (https://www.continuum.io/why-anaconda) or Enthought Canopy (https://www.enthought.com/products/canopy).

Questions, comments, suggestions, corrections, etc: email losc@ligo.caltech.edu

v20160208b

Intro to signal processing

This tutorial assumes that you know python well enough.
My Projects: Same Open Philosophy
Understanding Data Types in Python

Effective data-driven science and computation requires understanding how data is stored and manipulated. This section outlines and contrasts how arrays of data are handled in the Python language itself, and how NumPy improves on this. Understanding this difference is fundamental to understanding much of the material throughout the rest of the book.

Users of Python are often drawn-in by its ease of use, one piece of which is dynamic typing. While a statically-typed language like C or Java requires each variable to be explicitly declared, a dynamically-typed language like Python skips this specification. For example, in C you might specify a particular operation as follows:

```c
/* C code */
int result = 0;
for(int i=0; i<100; i++){
    result += i;
}
```

Entire content available on GitHub as Jupyter Notebooks
Python World Influencing Science . . .

Scientists are increasingly hosting research code on Github & similar services to aid in reproducibility.
## Python World Influencing Science . . .

Python’s software practices increasingly adopted by academia

<table>
<thead>
<tr>
<th>Traditional Astronomy Software</th>
<th>Python &amp; Open Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>Possessive/non-sharing</td>
<td>Cooperative/sharing</td>
</tr>
<tr>
<td>Fragmented &amp; Overlapping efforts</td>
<td>Build on common projects</td>
</tr>
<tr>
<td>Top-down planning</td>
<td>Bottom-up/Loose organization</td>
</tr>
<tr>
<td>Committee-oriented design</td>
<td>Design by “doers”</td>
</tr>
<tr>
<td>Endless analysis &amp; argument</td>
<td>Action-oriented &amp; experimentation</td>
</tr>
<tr>
<td>Unwilling to discard old tech</td>
<td>Good at replacing old tech</td>
</tr>
<tr>
<td>No leader to resolve conflicts</td>
<td>BDFL resolves conflicts</td>
</tr>
</tbody>
</table>

Adapted From Perry Greenfield’s PyData Keynote
Why is Python such an effective tool in science?

1. Interoperability with Other Languages
2. “Batteries Included” + Third-Party Modules
3. Simplicity & Dynamic Nature
4. Open ethos well-fit to science
Thank You!

Email: jakevdp@uw.edu

Twitter: @jakevdp

Github: jakevdp

Web: http://vanderplas.com/

Blog: http://jakevdp.github.io/