R vs. Python for Data Science?

Norm Matloff

Dept. of Computer Science
University of California, Davis

Invited Talk
SDSS 2020

URL for these slides:
`http://heather.cs.ucdavis.edu/RvsPythonForDS.pdf`

Norm Matloff

Dept. of Computer Science
University of California, Davis

Where I'm coming from

Norm Matloff

Dept. of Computer Science
University of California, Davis

Where I'm coming from

- User of both languages since near the beginning.

Where I'm coming from

- User of both languages since near the beginning.
- Former S user, transitioned early to R ("free S").

Norm Matloff

Dept. of Computer Science University of California, Davis

Where I'm coming from

- User of both languages since near the beginning.
- Former S user, transitioned early to R ("free S").
- Hated PERL, thus welcomed Python early in its development.

Norm Matloff

Dept. of Computer Science University of California, Davis

Where I'm coming from

- User of both languages since near the beginning.
- Former S user, transitioned early to R ("free S").
- Hated PERL, thus welcomed Python early in its development.
- Switched to Python for my admin tasks.

Norm Matloff

Dept. of Computer Science University of California, Davis

Where I'm coming from

- User of both languages since near the beginning.
- Former S user, transitioned early to R ("free S").
- Hated PERL, thus welcomed Python early in its development.
- Switched to Python for my admin tasks.
- Later, switched to R for my admin tasks, not just data science.

Norm Matloff

Dept. of Computer Science University of California, Davis

Where I'm coming from

- User of both languages since near the beginning.
- Former S user, transitioned early to R ("free S").
- Hated PERL, thus welcomed Python early in its development.
- Switched to Python for my admin tasks.
- Later, switched to R for my admin tasks, not just data science.
- Teach both languages.

Norm Matloff

Dept. of Computer Science University of California, Davis

Where I'm coming from

- User of both languages since near the beginning.
- Former S user, transitioned early to R ("free S").
- Hated PERL, thus welcomed Python early in its development.
- Switched to Python for my admin tasks.
- Later, switched to R for my admin tasks, not just data science.
- Teach both languages.
- Author of several books on, or using, R.

Norm Matloff

Dept. of Computer Science University of California, Davis

Where I'm coming from

- User of both languages since near the beginning.
- Former S user, transitioned early to R ("free S").
- Hated PERL, thus welcomed Python early in its development.
- Switched to Python for my admin tasks.
- Later, switched to R for my admin tasks, not just data science.
- Teach both languages.
- Author of several books on, or using, R.
- Former Editor-in-Chief, *The R Journal*.

Where I'm coming from

- User of both languages since near the beginning.
- Former S user, transitioned early to R ("free S").
- Hated PERL, thus welcomed Python early in its development.
- Switched to Python for my admin tasks.
- Later, switched to R for my admin tasks, not just data science.
- Teach both languages.
- Author of several books on, or using, R.
- Former Editor-in-Chief, *The R Journal*.

Thus a definite bias toward R, but am also an enthusiastic *Pythonista*.

Norm Matloff

Dept. of Computer Science
University of California, Davis

Overview

Norm Matloff

Dept. of Computer Science
University of California, Davis

Overview

Here I will argue in favor of R or Python on each of the below criteria.

Norm Matloff

Dept. of Computer Science University of California, Davis

Overview

Here I will argue in favor of R or Python on each of the below criteria. (If your favorite criterion is missing, please bring it up in Q&A.)

Norm Matloff

Dept. of Computer Science University of California, Davis

Overview

Here I will argue in favor of R or Python on each of the below criteria. (If your favorite criterion is missing, please bring it up in Q&A.)

- Elegance.
- Learning curve
- Available libraries for Data Science
- Machine learning
- Statistical sophistication
- Parallel computation
- C/C++ interface and performance enhancement
- Object orientation, metaprogramming
- Language unity
- Linked data structures
- Online help

Norm Matloff

Dept. of Computer Science
University of California, Davis

Elegance

Elegance

Clear win for Python.

Personally, I really appreciate Python's clean lines:

```
if x > y:
    z = 5
    w = 8
```

versus

```
if (x > y)
{
    z = 5
    w = 8
}
```

Python class structure cleaner than the various R structures.

Norm Matloff

Dept. of Computer Science
University of California, Davis

Learning curve

Learning curve

Huge win for R.

Learning curve

Huge win for R.

- I like to say,

  R was developed by statisticians for statisticians.

  (Replace *statisticians* by *data scientists* if you wish.)

Learning curve

Huge win for R.

- I like to say,

  R was developed by statisticians for statisticians.

  (Replace *statisticians* by *data scientists* if you wish.)
- But I'd also say,

  Python was developed by computer scientists for computer scientists.

Learning curve

Huge win for R.

- I like to say,

  R was developed by statisticians for statisticians.

  (Replace *statisticians* by *data scientists* if you wish.)
- But I'd also say,

  Python was developed by computer scientists for computer scientists.

- In Data Science, many people from backgrounds other than Computer Science or the like.

Learning curve

Huge win for R.

- I like to say,

  R was developed by statisticians for statisticians.

  (Replace *statisticians* by *data scientists* if you wish.)
- But I'd also say,

  Python was developed by computer scientists for computer scientists.

- In Data Science, many people from backgrounds other than Computer Science or the like.
- Python, especially in usage of libraries, really requires some computer systems sophistication.

Norm Matloff

Dept. of Computer Science
University of California, Davis

Learning curve cont'd.

Learning curve cont'd.

- Python libraries can be tricky to configure, even for the systems-savvy, while most R packages run right out of the box.

Learning curve cont'd.

- Python libraries can be tricky to configure, even for the systems-savvy, while most R packages run right out of the box.
- To even get started in Data Science with Python, one must learn a lot of material not in base Python, e.g., NumPy, Pandas and matplotlib.

Learning curve cont'd.

- Python libraries can be tricky to configure, even for the systems-savvy, while most R packages run right out of the box.
- To even get started in Data Science with Python, one must learn a lot of material not in base Python, e.g., NumPy, Pandas and matplotlib.
- By contrast, matrix types and basic graphics are built-in to base R.

Learning curve cont'd.

- Python libraries can be tricky to configure, even for the systems-savvy, while most R packages run right out of the box.
- To even get started in Data Science with Python, one must learn a lot of material not in base Python, e.g., NumPy, Pandas and matplotlib.
- By contrast, matrix types and basic graphics are built-in to base R. The novice can be doing simple data analyses within minutes.

Learning curve cont'd.

- Python libraries can be tricky to configure, even for the systems-savvy, while most R packages run right out of the box.
- To even get started in Data Science with Python, one must learn a lot of material not in base Python, e.g., NumPy, Pandas and matplotlib.
- By contrast, matrix types and basic graphics are built-in to base R. The novice can be doing simple data analyses within minutes.
- This alone should make Python a non-starter for Data Science.

Learning curve cont'd.

- Python libraries can be tricky to configure, even for the systems-savvy, while most R packages run right out of the box.
- To even get started in Data Science with Python, one must learn a lot of material not in base Python, e.g., NumPy, Pandas and matplotlib.
- By contrast, matrix types and basic graphics are built-in to base R. The novice can be doing simple data analyses within minutes.
- This alone should make Python a non-starter for Data Science.
- Of central importance, so I will elaborate here...

Norm Matloff

Dept. of Computer Science
University of California, Davis

Learning curve cont'd.

Norm Matloff

Dept. of Computer Science University of California, Davis

Learning curve cont'd.

Example, trying to install Keras on one of my machines:

```
Found existing installation: pip 8.1.1
Uninstalling pip-8.1.1:
Successfully uninstalled pip-8.1.1
Successfully installed pip-7.1.2
```

It took a *working version* of the package installer **pip** and inexplicably *uninstalled it*, replacing it with an *older* version! Even a systems-savvy person like me might have trouble tracking down the problem.

Norm Matloff

Dept. of Computer Science
University of California, Davis

Learning curve cont'd.

Learning curve cont'd.

As an example, I asked a Python sophisticate to install a library for PHATE, a visualization tool, thinking what a novice would see:

> *I tried it...using PyCharm...as IDE. I started off with a fresh install on a new computer, and I did run into some problems...Numpy.distutils.system_info.NotFoundError: No lapack/blas resources found...[the problem] after doing some google searching... is coming from some missing dependencies. According to stack overflow, one way around this is...*

He did say things went much better with Anaconda, but to me his experience epitomizes the problem:
Python is unnecessarily requiring too much expertise in the user.

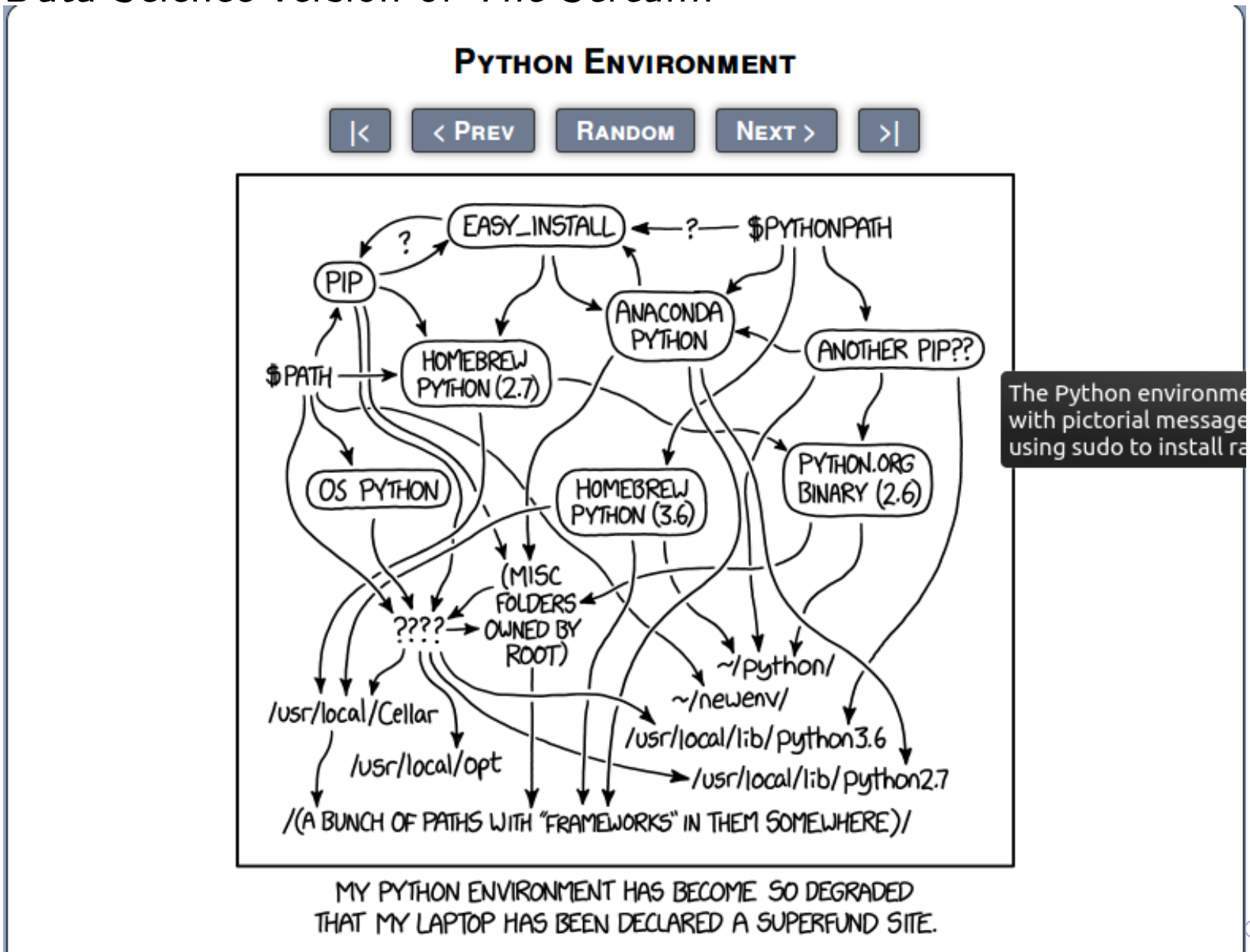
Norm Matloff

Dept. of Computer Science
University of California, Davis

Learning curve,xkcd

Learning curve,xkcd

Data Science version of *The Scream*:



Norm Matloff

Dept. of Computer Science
University of California, Davis

Available libraries for Data Science

Available libraries for Data Science

Slight edge to R.

Available libraries for Data Science

Slight edge to R.

- PyPI large but limited for data science.

Available libraries for Data Science

Slight edge to R.

- PyPI large but limited for data science.
- My (admittedly cursory) search for fast determination of nearest-neighbors on PyPI produced nothing. CRAN has at least 2 pkgs for R.

Available libraries for Data Science

Slight edge to R.

- PyPI large but limited for data science.
- My (admittedly cursory) search for fast determination of nearest-neighbors on PyPI produced nothing. CRAN has at least 2 pkgs for R.
- The following (again, cursory) searches in PyPI turned up nothing: *EM algorithm*; *log-linear model*; *Poisson regression*; *instrumental variables*; *spatial data*; *familywise error rate*

Available libraries for Data Science

Slight edge to R.

- PyPI large but limited for data science.
- My (admittedly cursory) search for fast determination of nearest-neighbors on PyPI produced nothing. CRAN has at least 2 pkgs for R.
- The following (again, cursory) searches in PyPI turned up nothing: *EM algorithm*; *log-linear model*; *Poisson regression*; *instrumental variables*; *spatial data*; *familywise error rate*

Norm Matloff

Dept. of Computer Science
University of California, Davis

Machine learning

Machine learning

Slight edge to Python.

Machine learning

Slight edge to Python.

- For many in ML, machine learning = neural networks.

Machine learning

Slight edge to Python.

- For many in ML, machine learning = neural networks.
- Since NNs have been developed mainly by CS people, the more sophisticated libraries, esp. for image classification, tend to be in Python.

Machine learning

Slight edge to Python.

- For many in ML, machine learning = neural networks.
- Since NNs have been developed mainly by CS people, the more sophisticated libraries, esp. for image classification, tend to be in Python.
- But random forests, gradient boosting etc. have been developed mainly by stat people, and R has excellent packages for these.

Machine learning

Slight edge to Python.

- For many in ML, machine learning = neural networks.
- Since NNs have been developed mainly by CS people, the more sophisticated libraries, esp. for image classification, tend to be in Python.
- But random forests, gradient boosting etc. have been developed mainly by stat people, and R has excellent packages for these.
- Want to do NNs in R? RStudio put in a huge effort to develop the R **keras** package, and it's excellent. H2O too.

Norm Matloff

Dept. of Computer Science
University of California, Davis

Statistical sophistication

Statistical sophistication

Big win for R.

Statistical sophistication

Big win for R.

- Again: R was developed *by* statisticians *for* statisticians.

Statistical sophistication

Big win for R.

- Again: R was developed *by* statisticians *for* statisticians.
- I find that Python ML people are more interested in the CS side of a method, e.g. fast sorting, than the probabilistic meaning of the model.

Statistical sophistication

Big win for R.

- Again: R was developed *by* statisticians *for* statisticians.
- I find that Python ML people are more interested in the CS side of a method, e.g. fast sorting, than the probabilistic meaning of the model.
- They tend to downplay the stat, and often don't understand it.

Statistical sophistication

Big win for R.

- Again: R was developed *by* statisticians *for* statisticians.
- I find that Python ML people are more interested in the CS side of a method, e.g. fast sorting, than the probabilistic meaning of the model.
- They tend to downplay the stat, and often don't understand it.
- I was appalled recently to see one of the most prominent ML people state in his book that standardizing the data to mean-0, variance-1 means one is assuming the data are Gaussian — absolutely false and misleading.

Norm Matloff

Dept. of Computer Science
University of California, Davis

Parallel computation

Parallel computation

Let's call it a tie.

Parallel computation

Let's call it a tie.

- Python **multiprocessing** package much improved from before.

Parallel computation

Let's call it a tie.

- Python **multiprocessing** package much improved from before.
- Python currently has better GPU access.

Parallel computation

Let's call it a tie.

- Python **multiprocessing** package much improved from before.
- Python currently has better GPU access.
- But still, R **parallel** package is much easier to use.

C/C++ interface and performance enhancement

C/C++ interface and performance enhancement

Slight win for R.

- Python has SWIG, PyPy, Cython, variants.

C/C++ interface and performance enhancement

Slight win for R.

- Python has SWIG, PyPy, Cython, variants.
- Lots of excitement about Pybind11.

C/C++ interface and performance enhancement

Slight win for R.

- Python has SWIG, PyPy, Cython, variants.
- Lots of excitement about Pybind11.
- But the versatility of R's Rccp is really much more powerful.

C/C++ interface and performance enhancement

Slight win for R.

- Python has SWIG, PyPy, Cython, variants.
- Lots of excitement about Pybind11.
- But the versatility of R's Rccp is really much more powerful.
- And R's new ALTREP has tremendous promise.

Norm Matloff

Dept. of Computer Science
University of California, Davis

Language unity

Norm Matloff

Dept. of Computer Science University of California, Davis

Language unity

- Python has now successfully accomplished transition from 2.7 to 3.x.

Language unity

- Python has now successfully accomplished transition from 2.7 to 3.x.
- By contrast, R is rapidly devolving into two mutually unintelligible dialects/communities, ordinary R and the Tidyverse.

Norm Matloff

Dept. of Computer Science University of California, Davis

Language unity

- Python has now successfully accomplished transition from 2.7 to 3.x.
- By contrast, R is rapidly devolving into two mutually unintelligible dialects/communities, ordinary R and the Tidyverse.
- To some degree, that split also falls along the lines of people who do statistics and those who view Data Science as graphics and data wrangling.

Language unity

- Python has now successfully accomplished transition from 2.7 to 3.x.
- By contrast, R is rapidly devolving into two mutually unintelligible dialects/communities, ordinary R and the Tidyverse.
- To some degree, that split also falls along the lines of people who do statistics and those who view Data Science as graphics and data wrangling.
- I'm a skeptic re Tidy (*http://github.com/matloff/TidyverseSkeptic*), but no matter what one's view is, this split is not good for R.

Linked data structures

Linked data structures

Win for Python.

Linked data structures

Win for Python.

- E.g. binary trees.
- Easy in Python, hard in R.
- Not common in Data Science.
- There is the R package **datastructures**.

Norm Matloff

Dept. of Computer Science
University of California, Davis

Online help

Online help

Big win for R.

Online help

Big win for R.

- R's **help()** generally more helpful than Python's.
- Also, **example()**, vignettes.
- Same for R's *generic* functions. When I'm using a new package, I know that I can probably use **print()**, **plot()**, **summary()**, and so on, while I am exploring.

Norm Matloff

Dept. of Computer Science
University of California, Davis

A small example

A small example

- OMSI online exam tool (*github.com/matloff/omsi*).
- Rather complex client/server app.
- Written by a highly talented team of students under my direction.
- I had them write the exam tool itself in Python, as I thought it would be easier to get top students who knew Python well.
- But I wrote the companion grading code, also rather complex, myself. And I wrote in R, my preference.
- Not a stat/Data Science app at all.
- Yet R was just as usable as Python in this app.
- Unlike some claims to the contrary, Yes, R in fact IS a "real" language!