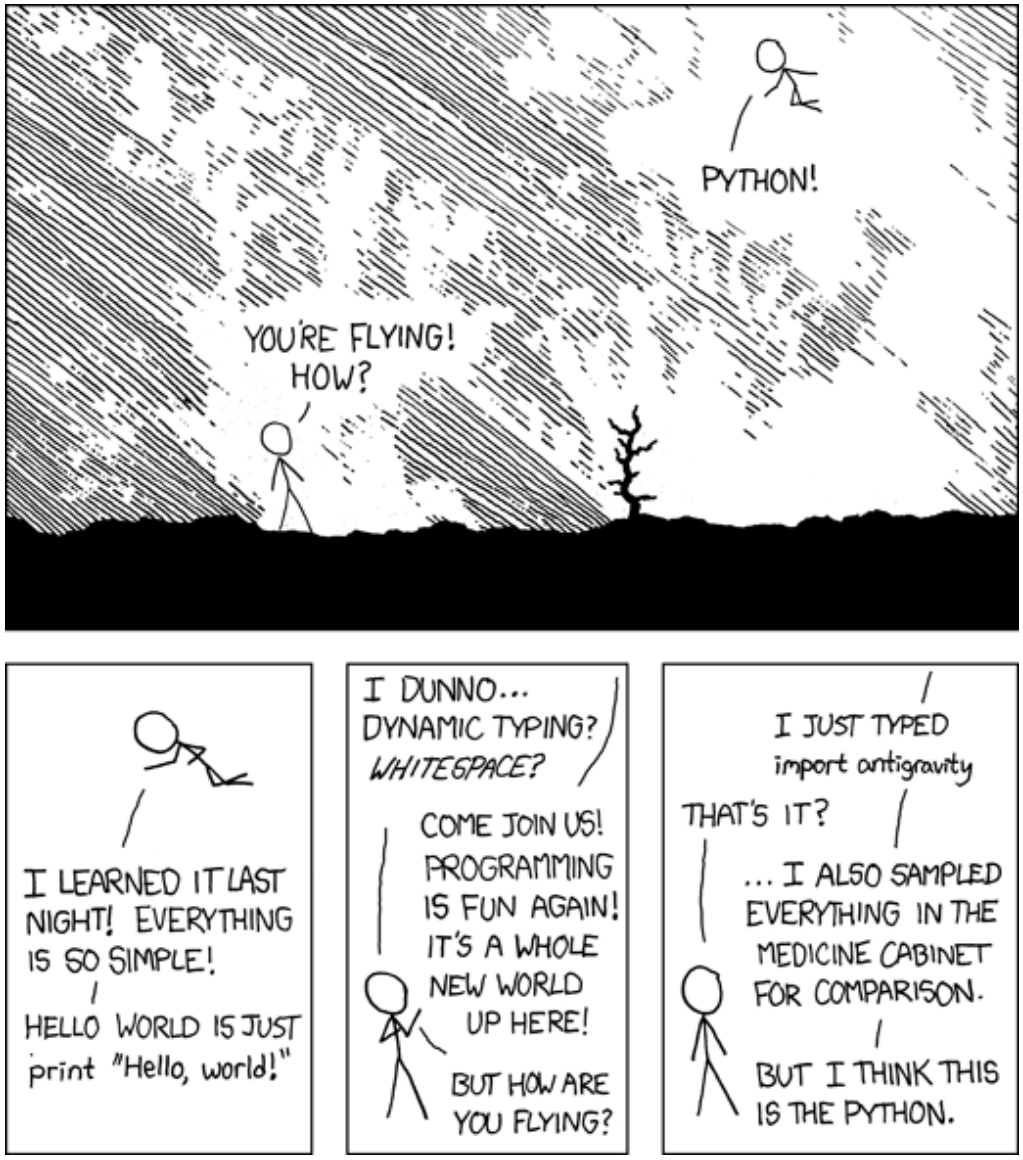
Programming with Python

EOAS Software Carpentry Workshop

September 24nd, 2015



https://xkcd.com/353

Getting started

For our Python introduction we're going to pretend to be a researcher named Harold Bridge (user id `hbridge`) who is studying inflammation in patients who have been given a new treatment for arthritis.

You need to download some files to follow this lesson:

1. Make a new folder in your Desktop called `python-novice-inflammation`.
2. Download python-novice-inflammation-data.zip and move the file to this folder.
3. If it's not unzipped yet, double-click on it to unzip it. You should end up with a new folder called data.
4. You can access this folder from the Unix shell with:

```
$ cd && cd Desktop/python-novice-inflammation/data
```

Launching Ipython (Jupyter) Notebook

There are several ways that we can use Python. We're going to start with a tool called Python Notebook that runs in the browser. In a shell window enter these commands:

```
$ cd
$ cd Desktop/python-novice-inflammation/data
$ ipython notebook
```

The shell window is now running a local web server for you. Don't close it. You will need to open another shell window to do other command line things. Your browser should open to an "Jupyter: Notebook" page showing a list of directories.

Analyzing patient data

1. Explain what a library is, and what libraries are used for.
2. Load a Python library and use the things it contains.
3. Read tabular data from a file into a program.
4. Assign values to variables.
5. Select individual values and subsections from data.

- import numpy
- numpy.loadtxt(fname= delimiter=)
- weight_kg = 55
- print('weight in kg:', weight_kg)
- weight_lb = 2.2 * weight_kg

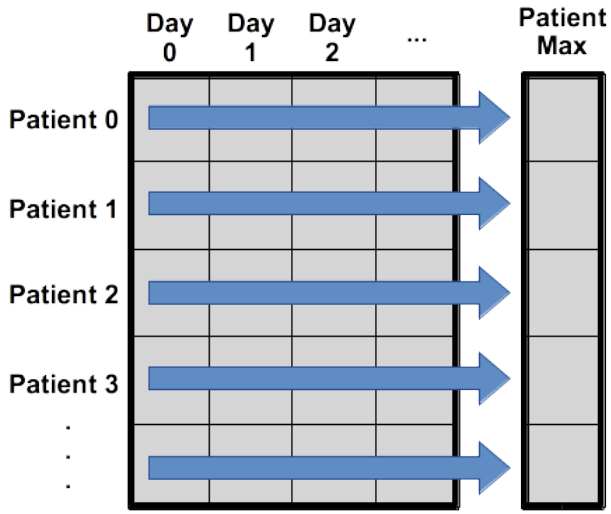
- type(data)
- data.shape
- data[0,0], data[0:1,0:1]
- data[0:10:2,1]
- data[:3,36:]

Analyzing Patient Data cont'd

6. Perform operations on arrays of data.
7. Display simple graphs.

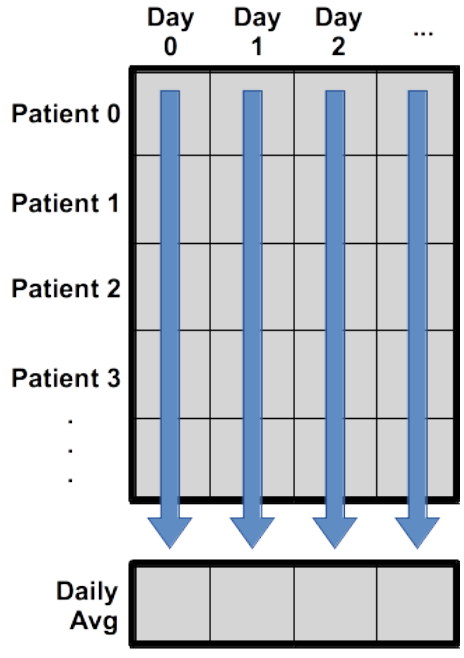
- data.mean()
- data.std()
- data.mean(axis=0)
- %matplotlib inline
- from matplotlib import pyplot
- pyplot.imshow(data)
- pyplot.show()
- pyplot.plot(ave_inflammation)
- import matplotlib import pyplot as plt
- plt.subplot(1,3,1)
- plt.ylabel('average')
- plt.show()

Operations across an axis



Max for each patient

`data.max(axis=1)`



Average for each day

`data.mean(axis=0)`

Exercise

Create a single plot showing 1) the mean for each day and 2) the mean + 1 standard deviation for each day and 3) the mean - 1 standard deviation for each day.

Repeating actions with loops

1. Explain what a for loop does.
2. Correctly write for loops to repeat simple calculations.
3. Trace changes to a loop variable as the loop runs.
4. Trace changes to other variables as they are updated by a for loop.

- for char in word:
- len('aeiou')

Python has a built-in function called range that creates a list of numbers: range(3) produces [0, 1, 2], range(2, 5) produces [2, 3, 4], and range(2, 10, 3) produces [2, 5, 8]. Using range, write a loop that prints the first three natural numbers:

```
1
2
3
```

Python has a built-in function called range that creates a list of numbers: range(3) produces [0, 1, 2], range(2, 5) produces [2, 3, 4], and range(2, 10, 3) produces [2, 5, 8]. Using range, write a loop that prints the first three natural numbers:

One solution:

```
for num in range(1,4,1):
 print(num)
```

Exponentiation is built into Python:

```
print(5**3)
125
```

Write a loop that calculates the same result using multiplication (without exponentiation).

Exponentiation is built into Python:

```
print(5**3)
125
```

Write a loop that calculates the same result using multiplication (without exponentiation)

One possible answer:

```
ans=1
for ii in range(1,4,1):
 ans=ans*5
print(ans)
```