Numerical Python

CS101 Lecture #15

Administrivia

- Homework #8 is due Friday, Dec. 2.

Numerical Python (`numpy`)

The problem

```
mydata = [ 4.5, 6.0, 1.2, 5.4 ]
from math import sin
sin(mydata)
```

- Why doesn't this work?
- `list` can contain any type!
- Also operators don't do what we "want":

```
mydata * 2.0  # doesn't double values!
```
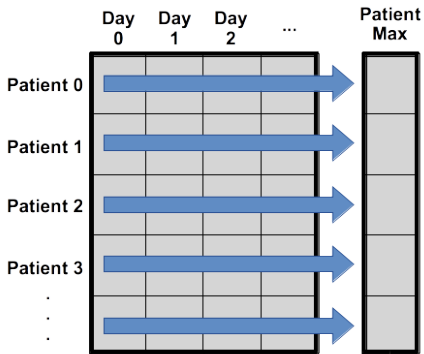
```
import numpy
import numpy as np  # better way
```

> ▸ numpy provides arrays and mathematical
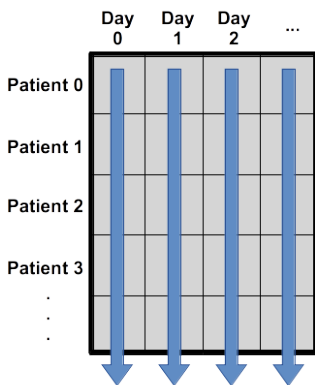>   functions.

```
data = np.array( [ 4.5, 6.0, 1.2, 5.4 ] )
data * 2.0
```

numpy

Consider a data set containing patient inflammation records for 60 patients over a period of 40 days, contained in `inflammation.csv`.

```
data = np.loadtxt( './data/inflammation.csv',
                   delimiter=',' )
```

Max for each patient
`data.max(axis=1)`

Average for each day
`data.mean(axis=0)`

Plotting (`matplotlib`)

```
import matplotlib.pyplot as plt
%matplotlib inline                          # jupyter only
```

- A plotting environment similar to MATLAB.
- Can plot `lists` or `arrays`.

```
xs = list( range(4) )
ys = [ 4.5, 6.0, 1.2, 5.4 ]
plt.plot( xs, ys )
plt.show()
```

matplotlib

- *Always* include labels:
- `plt.xlabel( 'domain' )`
- `plt.ylabel( 'range' )`
- `plt.title( 'topical data' )`

```
plt.plot( xs, ys )
plt.xlabel( 'x' )
plt.ylabel( 'y' )
plt.title( 'some values' )
plt.show()
```

matplotlib

- ❯ Basic cycle:
  - ▪ Add data to plot.
  - ▪ Add labels to plot.
  - ▪ Show plot.

matplotlib

- Two kinds of plots today:
- `plt.plot( x, y ) # for ptwise data`
- `plt.imshow( A ) # for array data`
- plot: third argument is *format string* (optional; color string + line style string); default as 'b-' for a solid blue line.

```
plt.plot( xs, ys, 'r.' )
plt.show()
```

- plot: can also take keyword arguments.

```
plt.plot( xs, ys, 'r.', label='height' )
plt.show()
```

Reminders

Reminders

- Homework #8 is due Friday, Dec. 2.