Lecture 9, September 14

Functions

Slides created by Ani Adhikari and John DeNero
Announcements

- **No late work.** If you joined the class late, please do current work. We’ll prorate based on when you joined.

- **Technical problems? Questions about hw/lab credit?** Please email your GSI, not me. Addresses are on the Staff Contact page in data8.org.

- **Python Playground: NEW!** There’s a link on Piazza. A place for you to experiment with Python and the class datasets.

- **Please fill out the Welcome Survey!**
Results so far: Requirement?

Does this course currently fulfill a requirement for one of your intended degree programs?

(345 responses)

- Yes: 63.2%
- Maybe: 10.1%
- No: 20%
- I don't know
Good Programmer?

How good a programmer do you consider yourself to be? (345 responses)

- A very good programmer: 38%
- A good programmer: 9.9%
- A reasonable programmer: 10.1%
- A bad programmer: 8.1%
- A terrible programmer: 29.3%
- I have no skill at programming: 10.1%
- I don't know: 9.9%
Defining Functions

- You get to write your own functions.
- Useful when you want to do the same computation over and over again, and Python doesn’t already have a function for it.

(Demo)
User-defined functions give names to blocks of code

```python
def spread(values):
    return max(values) - min(values)
```

(Demo)
Discussion Question

What does this function do? What kind of input does it take? What output will it give? What's a reasonable name?

def f(s):
    return np.round(s / sum(s) * 100, 2)
Apply

The **apply** method creates an array by calling a function on every element in an input column.

- First argument: Function to apply
- Second argument: The input column label

\[
\text{table\_name.apply(\text{function\_name}, \text\{\text{column\_label}\})}
\]

(Demo)