



Week 1

INTRODUCTION TO DATA SCIENCE IN PYTHON

M

Python

- Why Python?
 - It's easy to learn
 - Now the language of choice for 8 of 10 top US computer science programs (Philip Guo, CACM)
 - Full featured
 - Not just a statistics language, but has full capabilities for data acquisition, cleaning, databases, high performance computing, and more
 - Strong Data Science Libraries
 - The SciPy Ecosystem

Week 1

INTRODUCTION TO DATA SCIENCE IN PYTHON

M

Course Outline

- Prerequisite Python Knowledge
- The *pandas* Toolkit
- Advanced Querying and Manipulation with *pandas*
- Basic Statistical Analysis with *numpy* and *scipy*, and project

Week 1

INTRODUCTION TO DATA SCIENCE IN PYTHON

M

Data Science

- Drew Conway perspective on data science:
 - Hacking Skills
 - Math and Statistics Knowledge
 - Substantive Expertise
- Other data science perspectives:
 - Skepticism, experimentation, simulation, and replication

a good data scientist
• can communicate their findings clearly to others
• have fundamental tools in toolkit

Week 1

INTRODUCTION TO DATA SCIENCE IN PYTHON

M

Data Science

- David Donoho, “50 Years of Data Science”
 - Data Exploration and Preparation (cleaning data, manipulate)
 - Data Representation and Transformation (several forms of representing data (tabular, graph-based data))
 - Computing with Data (R & Python)
 - Data Modeling (pipelining - data scientists work with different languages for different parts of an analysis project)
 - Data Visualization and Presentation
 - Science about Data Science
- predicting modeling
- understand what works, what doesn't
 - build ways to leverage discovery

Modern data science projects span different languages
know when to use the right tool for the job

Week 1

INTRODUCTION TO DATA SCIENCE IN PYTHON

M

The map() function

```
map(function, iterable, ...)
```

Return an iterator that applies *function* to every item of *iterable*, yielding the results. If additional *iterable* arguments are passed, *function* must take that many arguments and is applied to the items from all iterables in parallel. With multiple iterables, the iterator stops when the shortest iterable is exhausted. For cases where the function inputs are already arranged into argument tuples, see `itertools.starmap()`.

1st parameter - want to executed, all iterable arguments
2nd -> param.: st iterated upon
unpacked together
pass into function