# Python

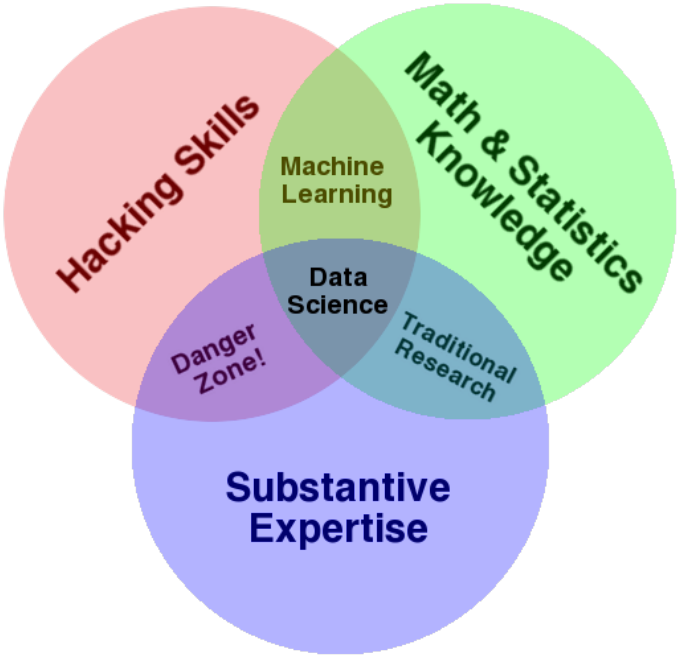
- **Why Python?**
  1. *It's easy to learn*
     - *Now the language of choice for 8 of 10 top US computer science programs (Philip Guo, CACM)*
  2. *Full featured*
     - *Not just a statistics language, but has full capabilities for data acquisition, cleaning, databases, high performance computing,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

# Course Outline

1. Prerequisite Python Knowledge
2. The *pandas* Toolkit
3. Advanced Querying and Manipulation with *pandas*
4. Basic Statistical Analysis with *numpy* and *scipy*, and project

# 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*



# Data Science

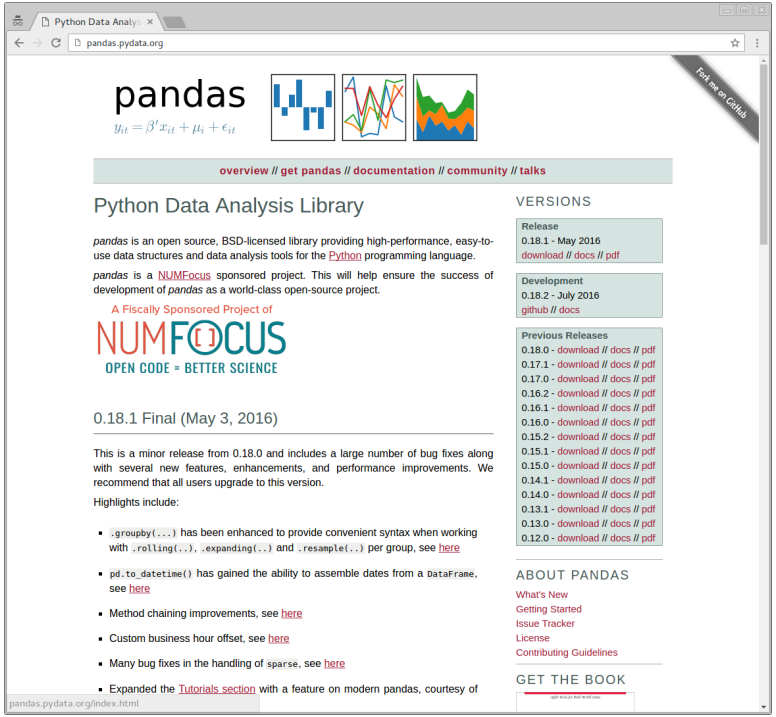
- **David Donoho, "50 Years of Data Science"**
  1. *Data Exploration and Preparation*
  2. *Data Representation and Transformation*
  3. *Computing with Data*
  4. *Data Modeling*
  5. *Data Visualization and Presentation*
  6. *Science about Data Science*

# The map() function

**map**(*function*, *iterable*, ...)

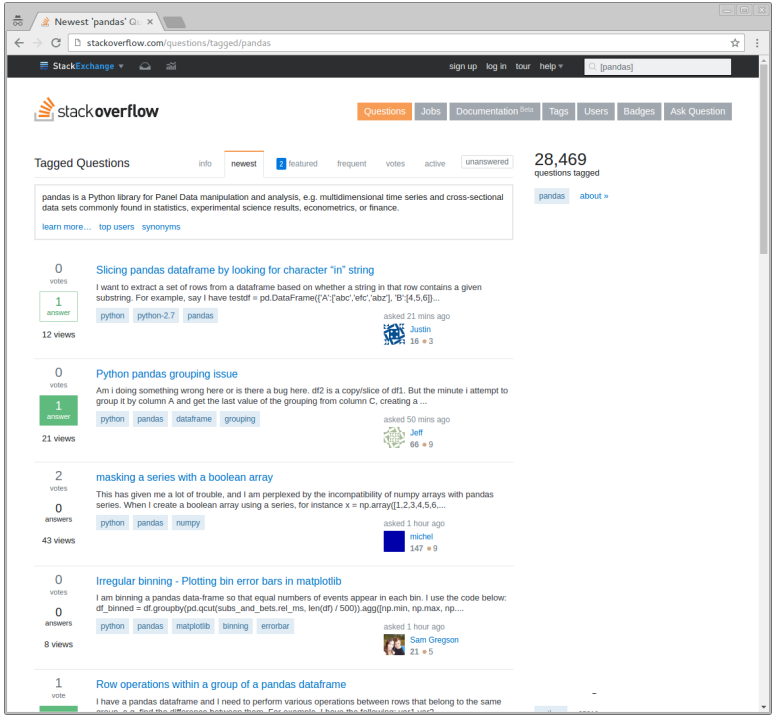
Return an iterator that applies *function* to every item of *iterable*, yielding the results. If additional *iterable* arguments are passed, *function* must take that many arguments and is applied to the items from all iterables in parallel. With multiple iterables, the iterator stops when the shortest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

# Pandas

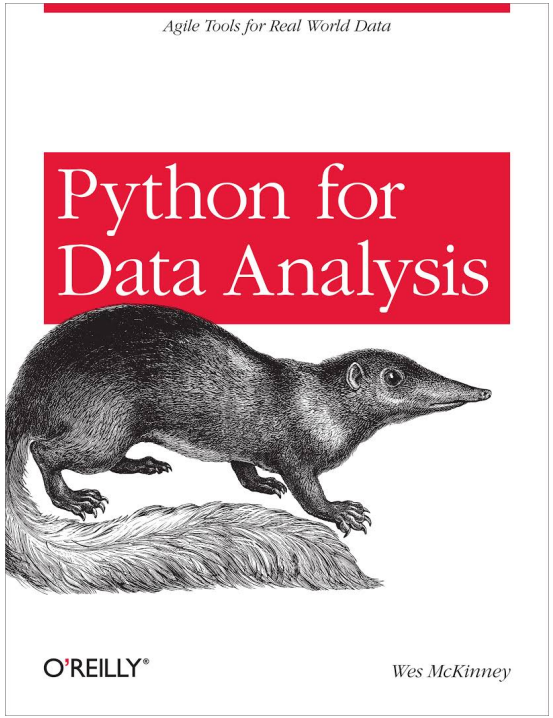


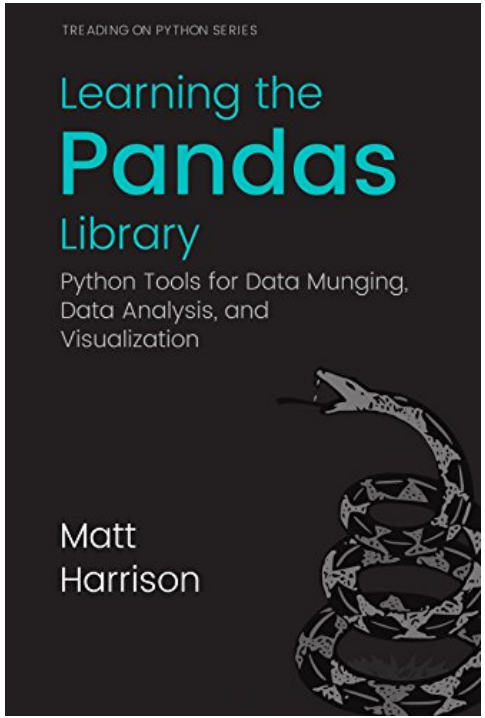

- Created in 2008 by Wes McKinney
- Open source New BSD license
- 100 different contributors

# Stack Overflow

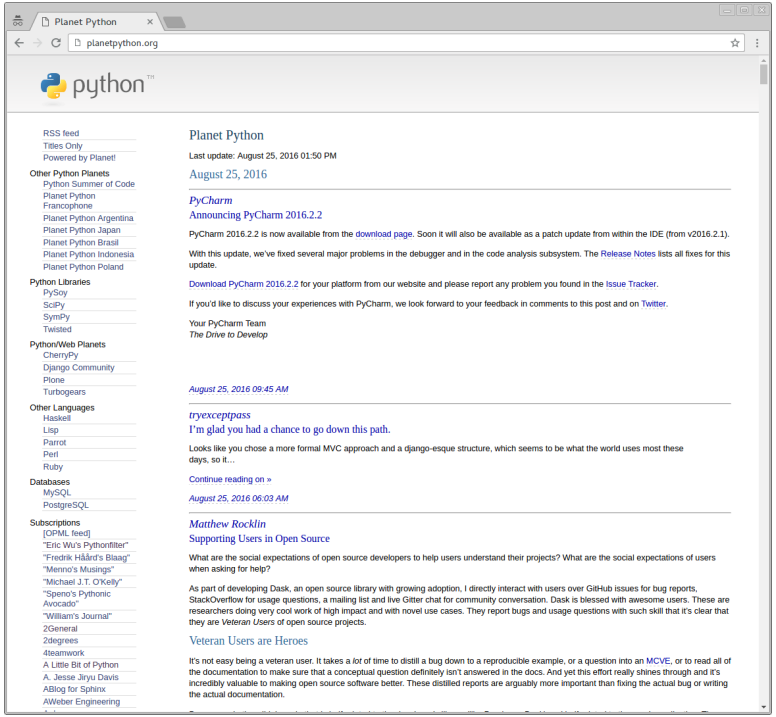



- http://stackoverflow.com
- Massive knowledge forum of python and pandas related content
- Free to join and participate in
- Heavily used by pandas developers instead of a mailing list


Agile Tools for Real World Data
Python for Data Analysis
O'REILLY
Wes McKinney

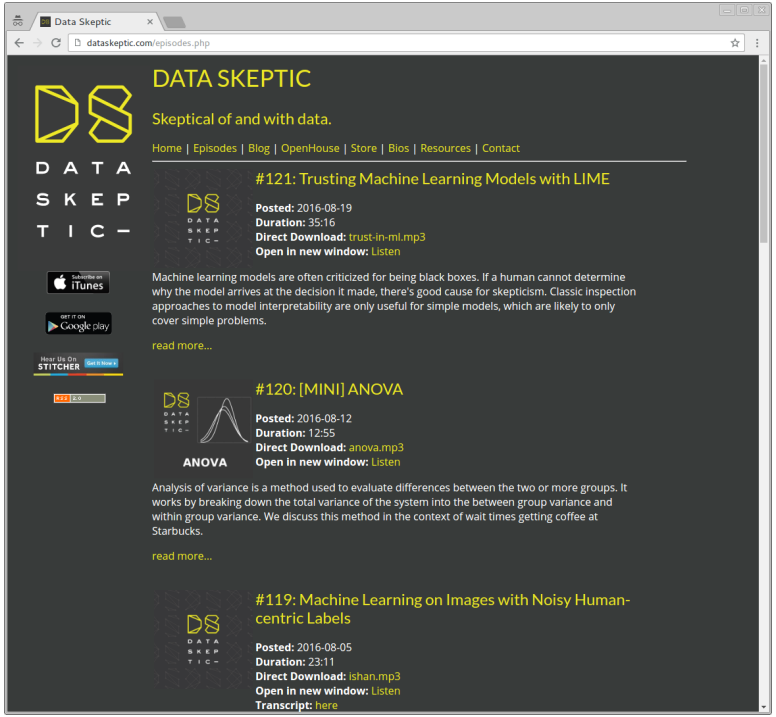

TREADING ON PYTHON SERIES
Learning the Pandas Library
Python Tools for Data Munging, Data Analysis, and Visualization
Matt Harrison

# Planet Python




- http://planetpython.org/
- Excellent blog aggregator for python related news
- Significant number of data science and python tutorials are posted
- Great blend of applied beginner and higher level python postings

# Data Skeptic Podcast




- http://dataskeptic.com/
- Kyle Polich, created in 2014
- Covers data science more generally, including:
  - *Mini educational lessons*
  - *Interviews*
  - *Trends*
  - *Shared community project (OpenHouse)*

# The Series

Name

Index

Values

| | Animals |
|---|---|
| 0 | Dog |
| 1 | Bear |
| 2 | Tiger |
| 3 | Moose |
| 4 | Giraffe |
| 5 | Hippopotamus |
| 6 | Mouse |

# The DataFrame

Index
.loc(), .iloc()

Axis 1 (columns)

.columns()

Axis 0 (rows)

| | Animals | Owners |
|---|---|---|
| 0 | Dog | Chris |
| 1 | Bear | Kevyn |
| 2 | Tiger | Bob |
| 3 | Moose | Vinod |
| 4 | Giraffe | Daniel |
| 5 | Hippopotamus | Fil |
| 6 | Mouse | Stephanie |

`df.iloc(2)`

Values

`df["Owners"]`

`df.iloc(5)["Animals"]`

W All-time Olympic Ga

https://en.wikipedia.org/wiki/All-time_Olympic_Games_medal_table

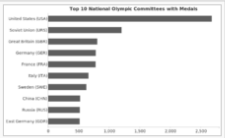
Čeština
Dansk
Deutsch
Español
Esperanto
فارسی
Français
한국어
Hrvatski
Bahasa Indonesia
Italiano
עברית
Latviešu
Magyar
Македонски
मराठी
Nederlands
日本語
Norsk bokmål
Occitan
ਪੰਜਾਬੀ
Polski
Português
Română
Русский
Српски / srpski
Suomi
Svenska
Tagalog
ไทย
Türkçe
Українська

## NOCs with medals [ edit ]

The silver medal was awarded to the winner of each event during the 1896 Summer Olympics. The current system of gold, silver, and bronze medals was not implemented until the 1912 Olympic Games

The table is pre-sorted by the name of each Olympic Committee, but can be displayed as sorted by any other column, such as the total number of gold medals or total number of overall medals. To sort by gold, silver, and then bronze, sort first by the bronze column, then the silver, and then the gold. The table does not include the medals revoked (e.g., due to doping, etc.). Medal totals in this table are current as of the 2014 Winter Olympics in Sochi, and all changes in medal standings due to doping cases up to and including 10 November 2015 are taken into account.

| Team (IOC code) | № Summer | 1 | 2 | 3 | Total | № Winter | 1 | 2 | 3 | Total | № Games | 1 | 2 | 3 | Combined total |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Afghanistan (AFG) | 13 | 0 | 0 | 2 | 2 | 0 | 0 | 0 | 0 | 0 | 13 | 0 | 0 | 2 | 2 |
| Algeria (ALG) | 12 | 5 | 2 | 8 | 15 | 3 | 0 | 0 | 0 | 0 | 15 | 5 | 2 | 8 | 15 |
| Argentina (ARG) | 23 | 18 | 24 | 28 | 70 | 18 | 0 | 0 | 0 | 0 | 41 | 18 | 24 | 28 | 70 |
| Armenia (ARM) | 5 | 1 | 2 | 9 | 12 | 6 | 0 | 0 | 0 | 0 | 11 | 1 | 2 | 9 | 12 |
| *Australasia (ANZ)* [ANZ] | 2 | 3 | 4 | 5 | 12 | 0 | 0 | 0 | 0 | 0 | 2 | 3 | 4 | 5 | 12 |
| Australia (AUS) [AUS] [Z] | 25 | 139 | 152 | 177 | 468 | 18 | 5 | 3 | 4 | 12 | 43 | 144 | 155 | 181 | 480 |
| Austria (AUT) | 26 | 18 | 33 | 35 | 86 | 22 | 59 | 78 | 81 | 218 | 48 | 77 | 111 | 116 | 304 |
| Azerbaijan (AZE) | 5 | 6 | 5 | 15 | 26 | 5 | 0 | 0 | 0 | 0 | 10 | 6 | 5 | 15 | 26 |
| Bahamas (BAH) | 15 | 5 | 2 | 5 | 12 | 0 | 0 | 0 | 0 | 0 | 15 | 5 | 2 | 5 | 12 |
| Bahrain (BRN) | 8 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 8 | 0 | 0 | 1 | 1 |
| Barbados (BAR) [BAR] | 11 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 11 | 0 | 0 | 1 | 1 |
| Belarus (BLR) | 5 | 12 | 24 | 39 | 75 | 6 | 6 | 4 | 5 | 15 | 11 | 18 | 28 | 44 | 90 |
| Belgium (BEL) | 25 | 37 | 52 | 53 | 142 | 20 | 1 | 1 | 3 | 5 | 45 | 38 | 53 | 56 | 147 |
| Bermuda (BER) | 17 | 0 | 0 | 1 | 1 | 7 | 0 | 0 | 0 | 0 | 24 | 0 | 0 | 1 | 1 |




Top 10 National Olympic Committees with medals

df

| | Animals | Owners |
|---|---|---|
| 0 | Dog | Chris |
| 1 | Bear | Kevyn |
| 2 | Tiger | Bob |
| 3 | Moose | Vinod |
| 4 | Giraffe | Daniel |
| 5 | Hippo | Fil |
| 6 | Mouse | Stephanie |

+

Boolean mask

| | |
|---|---|
| True | True |
| True | True |
| False | False |
| True | True |
| False | False |
| False | False |
| False | False |

=

result

| | Animals | Owners |
|---|---|---|
| 0 | Dog | Chris |
| 1 | Bear | Kevyn |
| 3 | Moose | Vinod |

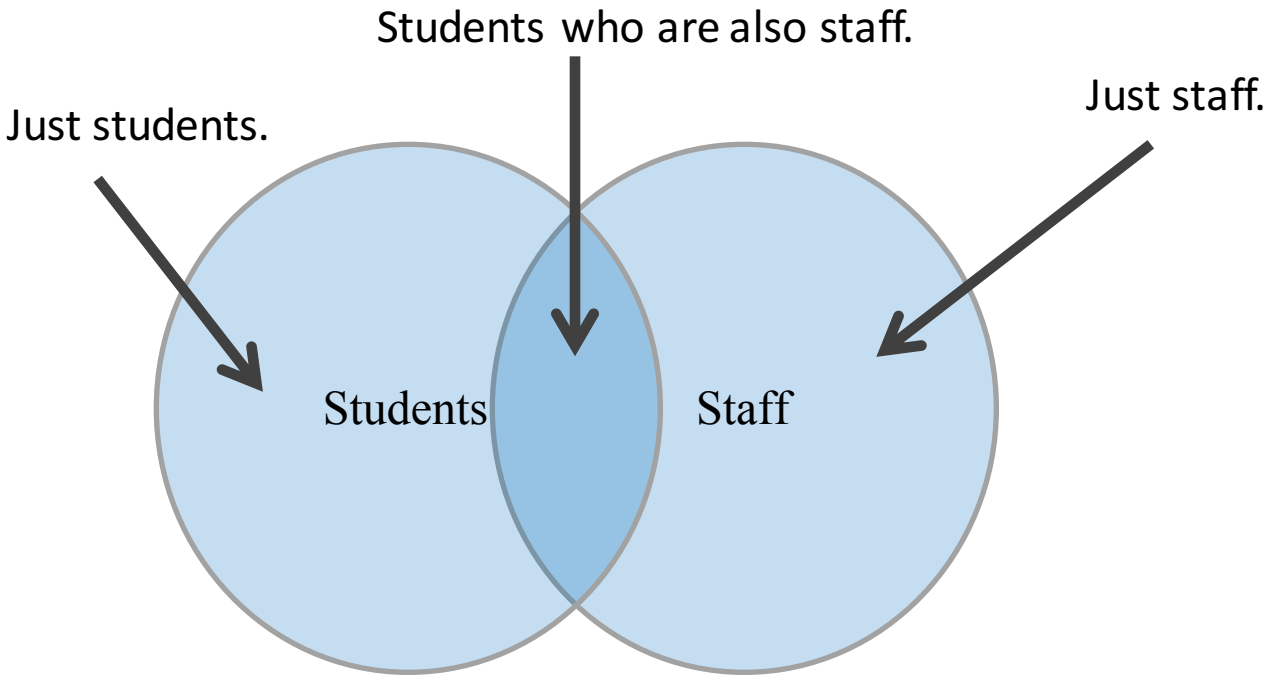
# Pandas Data Structures

- **Series Object (1 dimensional, a row)**
- **DataFrame Object (2 dimensional, a table)**
- **Querying**
  - *iloc[], for querying based on <u>position</u>*
  - *loc[], for querying rows based on <u>label</u>*
  - *Querying the DataFrame directly*
    - *Projecting a subset of columns*
    - *Using a boolean mask to filter data*

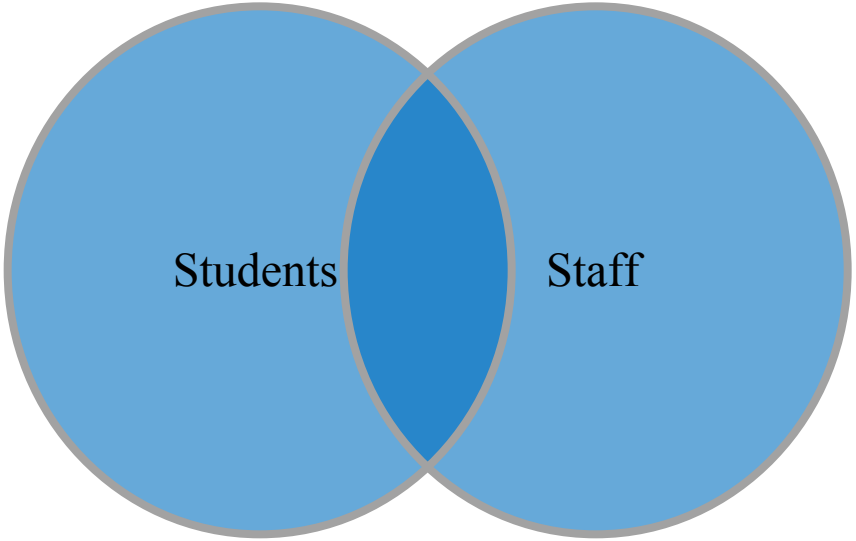
# Setting Data in Pandas

- **To add new data**
  - `df[column]=[a,b,c]`
- **To set default data (or overwrite all data):**
  - `df[column]=2`

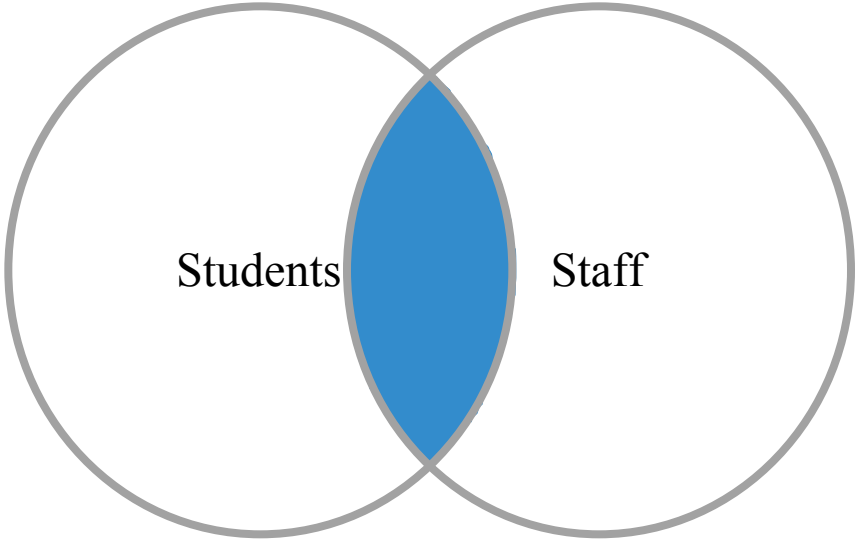
# Venn Diagram



# Full outer join (union)



# Inner join (intersection)



- **Chain Indexing:**
  - *df.loc["Washtenaw"]["Total Population"]*
  - *Generally bad, pandas could return a copy of a view depending upon numpy*
- **Code smell**
  - *If you see a ][ you should think carefully about what you are doing (Tom Augspurger)*

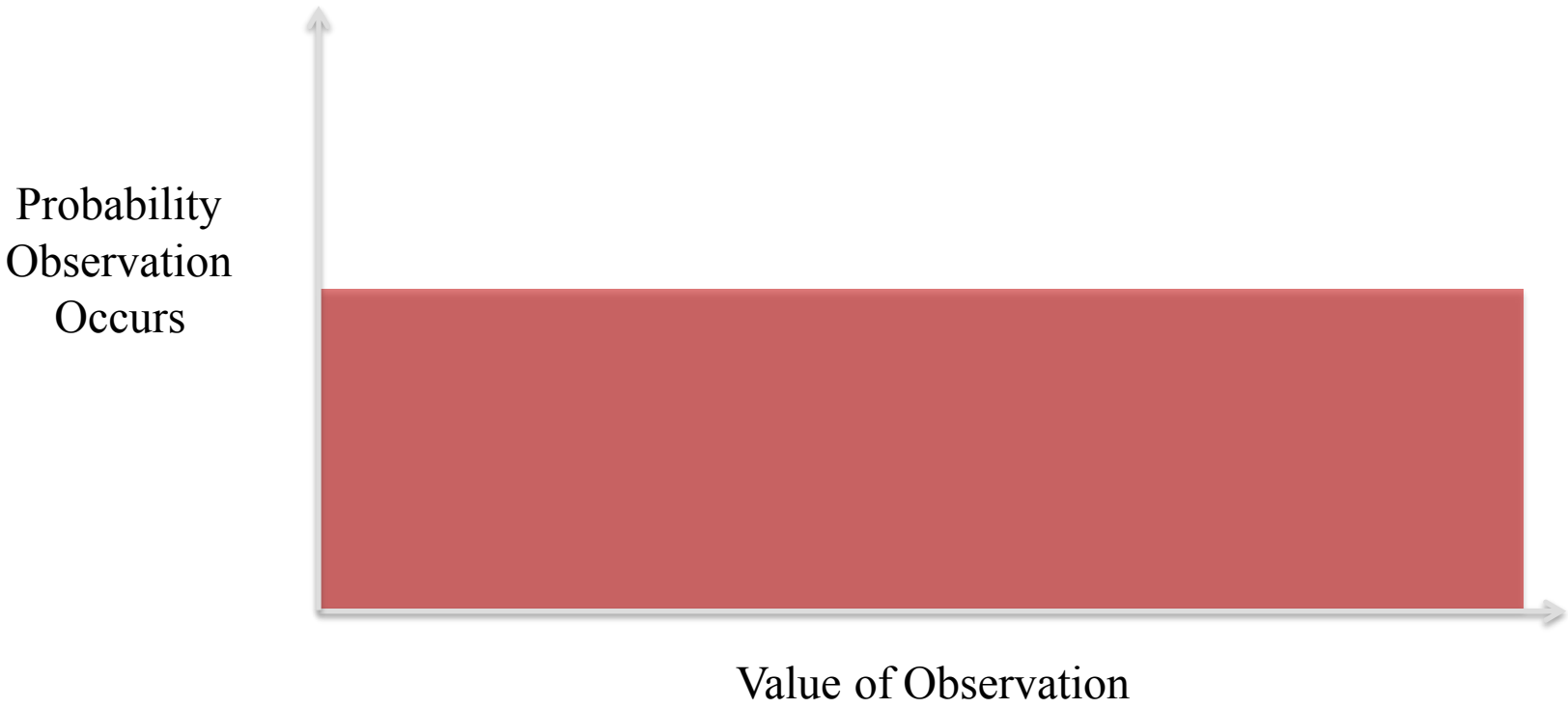
# (a,b) (c,d): Scales

- **Ratio scale:**
  - *units are equally spaced*
  - *mathematical operations of +-/* are all valid*
  - *E.g. height and weight*
- **Interval scale:**
  - *units are equally spaced, but there is no true zero*
- **Ordinal scale:**
  - *the order of the units is important, but not evenly spaced.*
  - *Letter grades such as A+, A are a good example*
- **Nominal scale:**
  - *categories of data, but the categories have no order with respect to one another.*
  - *E.g. Teams of a sport.*

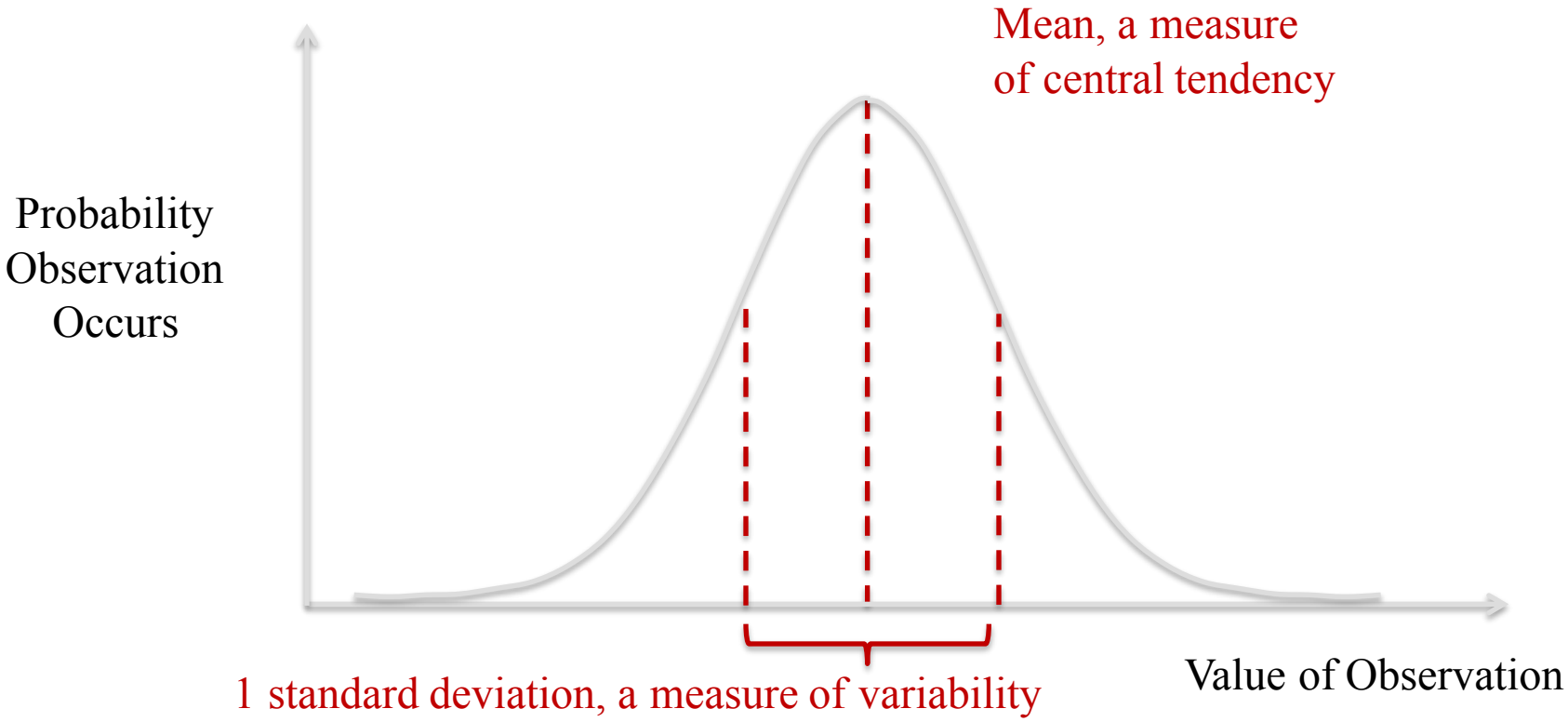
# Distributions

- **Distribution: Set of all possible random variables**
- **Example:**
  - *Flipping Coins for heads and tails*
    - *a binomial distribution (two possible outcomes)*
    - *discrete (categories of heads and tails, no real numbers)*
    - *evenly weighted (heads are just as likely as tails)*
  - *Tornado events in Ann Arbor*
    - *a binomial distribution*
    - *Discrete*
    - *evenly weighted (tornadoes are rare events)*

# Uniform Distribution (Continuous)

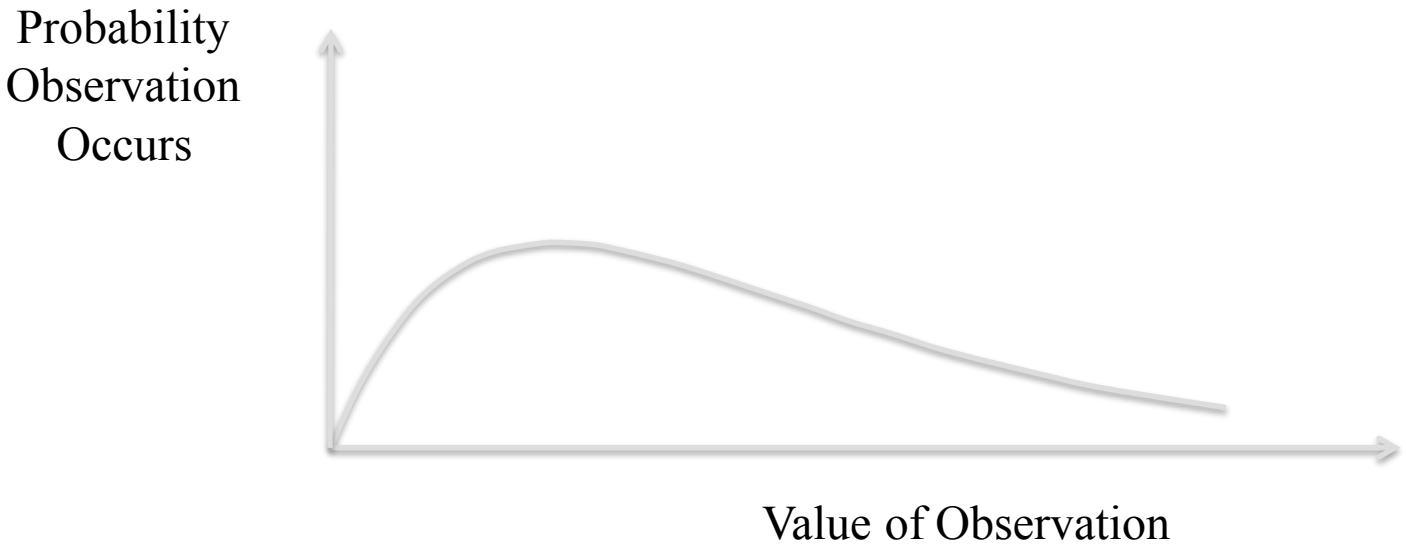


# Normal (Gaussian) Distribution

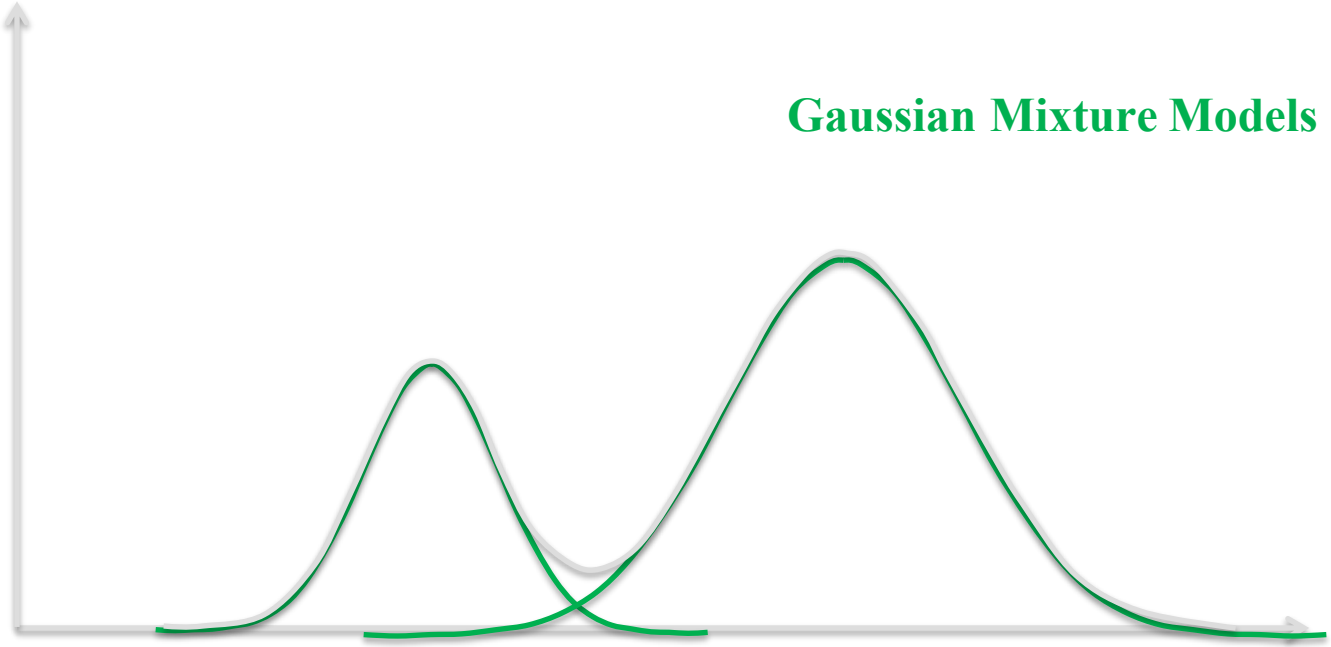


# Chi Squared ($\chi^2$) Distribution

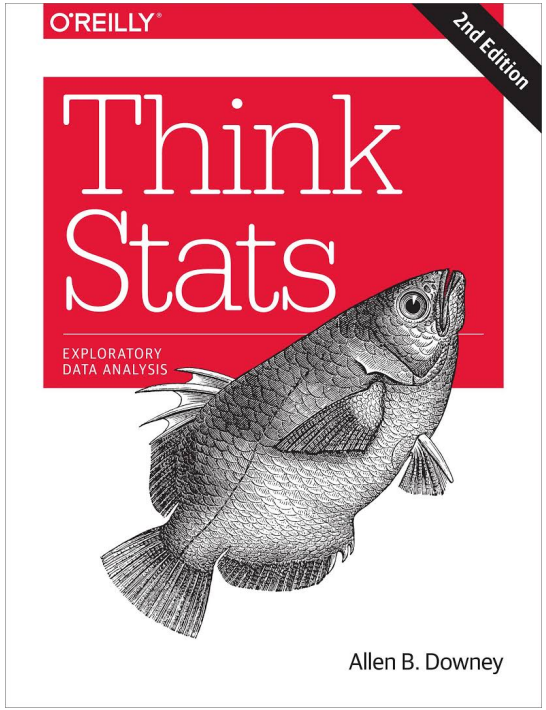
- **Left-skewed**
- **Degrees of freedom = 4**



# Bimodal distributions



# Think Stats




- **Probability and Statistics for Programmers**
  - *Allen B. Downey*
  - *Available for free under CC license at:*

*http://greenteapress.com/thinkstats2/index.html*

# Hypothesis Testing

- **Hypothesis: A statement we can test**
  - *Alternative hypothesis: Our idea, e.g. there is a difference between groups*
  - *Null hypothesis: The alternative of our idea, e.g. there is no difference between groups*
- **Critical Value alpha (α)**
  - *The threshold as to how much chance you are willing to accept*
  - *Typical values in social sciences are 0.1, 0.05, or 0.01*

# p-hacking

- **P-hacking, or Dredging**
  - *Doing many tests until you find one which is of statistical significance*
  - *At a confidence level of 0.05, we expect to find one positive result 1 time out of 20 tests*
  - *Remedies:*
    - *Bonferroni correction*
    - *Hold-out sets*
    - *Investigation pre-registration*