

What is Data Science?

and how to learn it

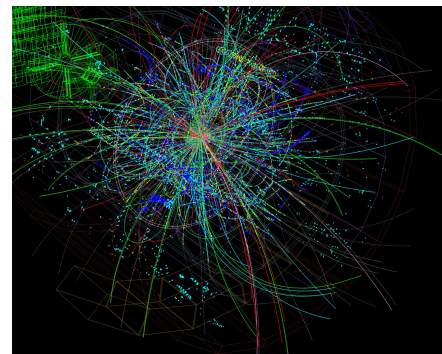
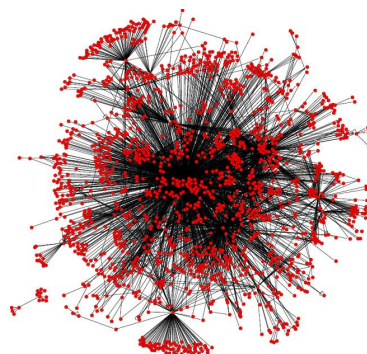
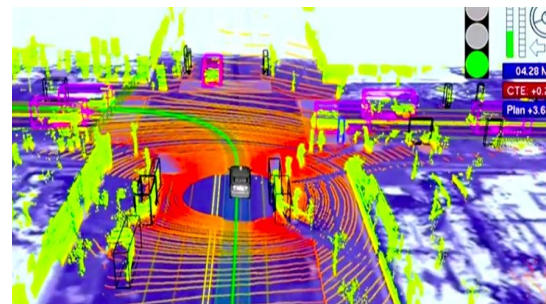
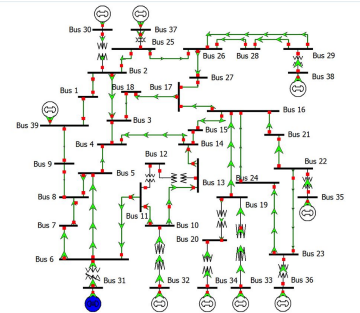
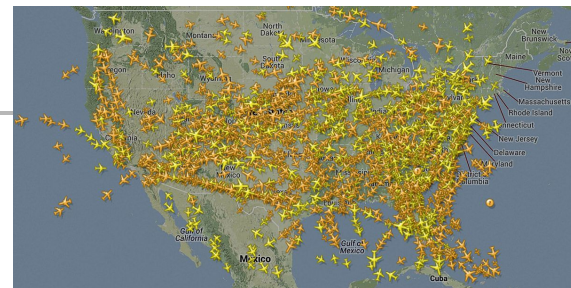
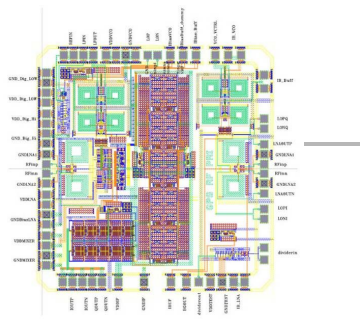
Examples

- Engineering

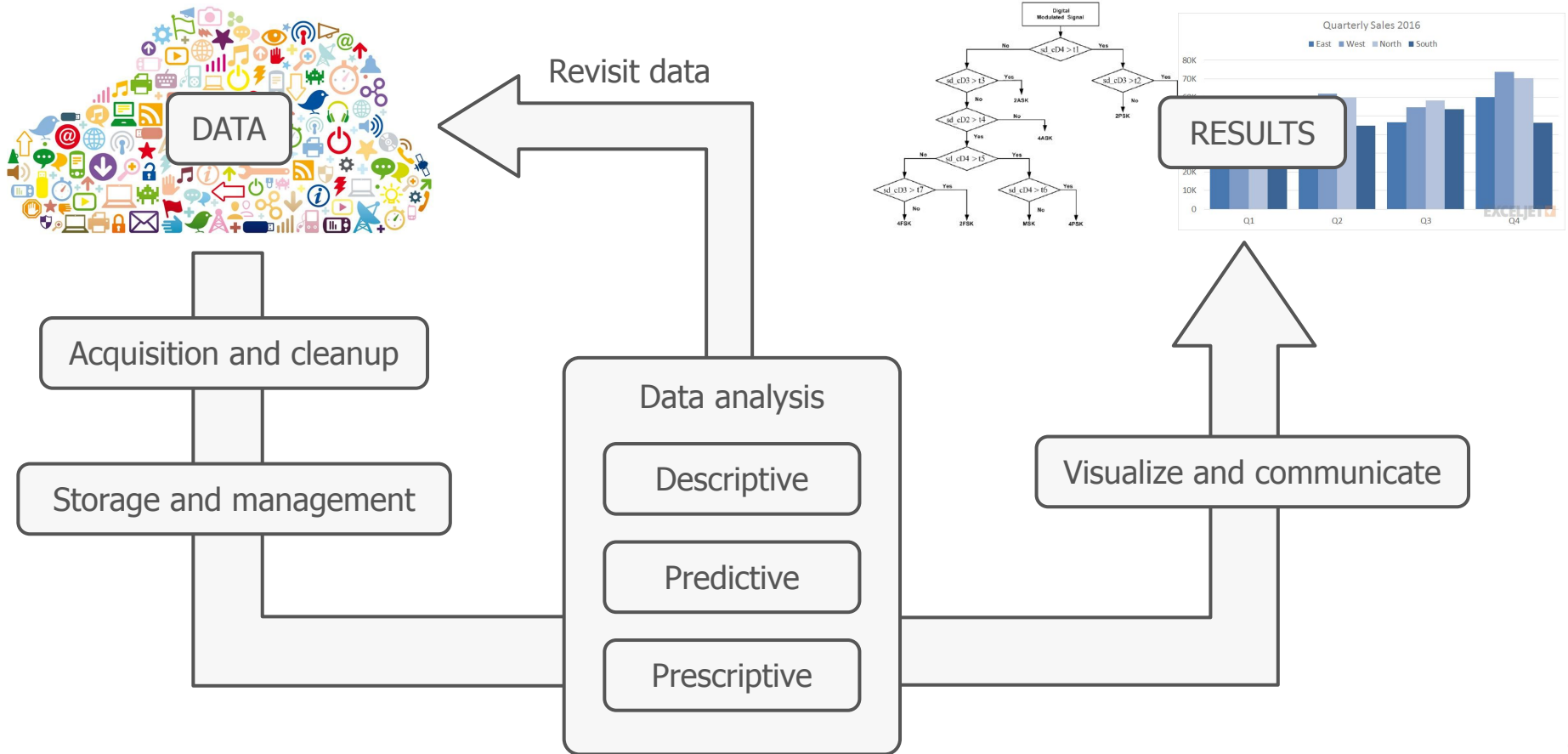
- Infrastructure and logistics
- Power systems
- Wireless telecommunication
- Robotics and manufacturing
- Chip design and optimization
- Autonomous vehicles

- Science

- Particle physics
- Biochemistry
- Neuroscience



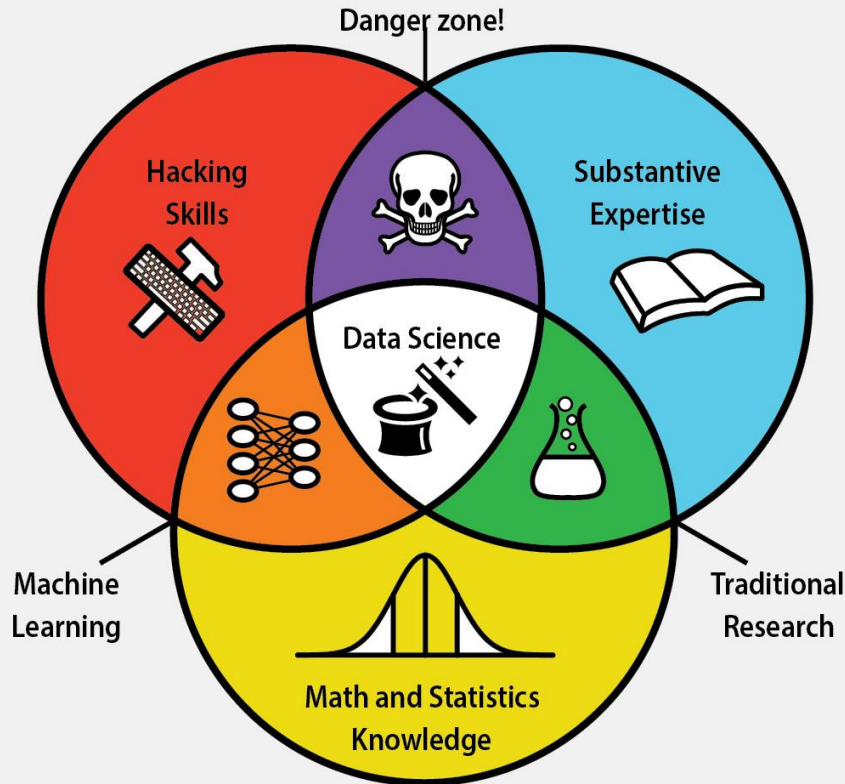
What is data science?



What's in a name?

- **Data science (DS)**: very broad; everything on the prev. slide!
- **Machine learning (ML)**: everything in the “data analysis” box. Usually used to refer to specific mathematical techniques.
- **Deep learning (DL)**: specific kind of machine learning model used in applications such as image recognition, speech recognition, sentiment analysis, and more.
- **Artificial intelligence (AI)**: very broad; any software designed to make a computer think/behave/learn like a human.

DATA SCIENCE SKILLSET



Data science, due to its interdisciplinary nature, requires an intersection of abilities: **hacking skills**, **math and statistics knowledge**, and **substantive expertise** in a field of science.



Hacking skills are necessary for working with massive amounts of electronic data that must be acquired, cleaned, and manipulated.



Math and statistics knowledge allows a data scientist to choose appropriate methods and tools in order to extract insight from data.



Substantive expertise in a scientific field is crucial for generating motivating questions and hypotheses and interpreting results.



Traditional research lies at the intersection of knowledge of math and statistics with substantive expertise in a scientific field.



Machine learning stems from combining hacking skills with math and statistics knowledge, but does not require scientific motivation.



Danger zone! Hacking skills combined with substantive scientific expertise without rigorous methods can beget incorrect analyses.

Overview of class topics

- “Hacking” skills
 - Coding in Python
 - Using Jupyter Notebooks
 - Managing messy data
 - “Learn to learn”
- Avoiding the danger zone
 - Conceptual and practical pitfalls
 - Bias, privacy, ethical issues
 - Stuff not in the flowchart diagram!
- Descriptive analysis
 - Finding structure in data
 - Communicating/visualizing data
- Predictive analysis
 - Modeling techniques
 - Supervised learning
- Case studies
 - Work with real data
 - Domain-specific issues

Tools and technologies



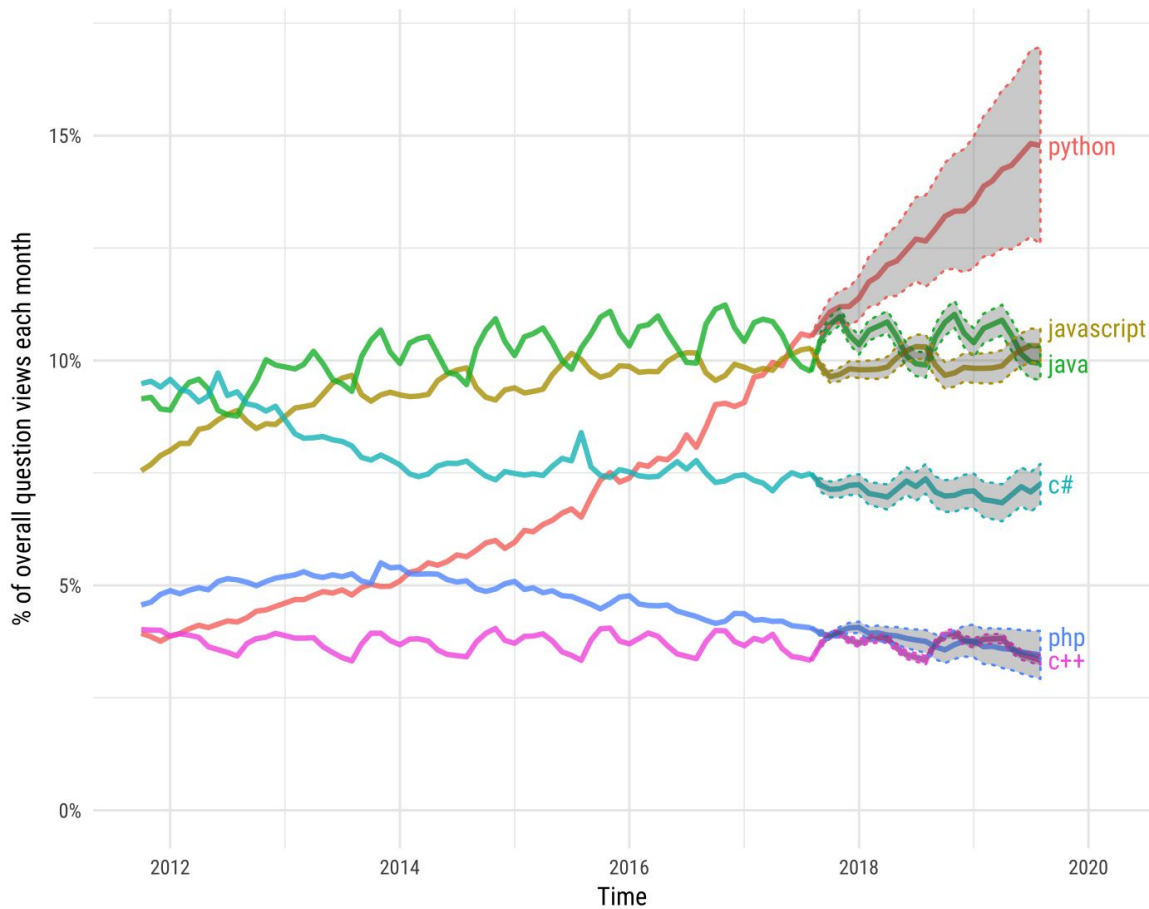
IP[y]: IPython
Interactive Computing



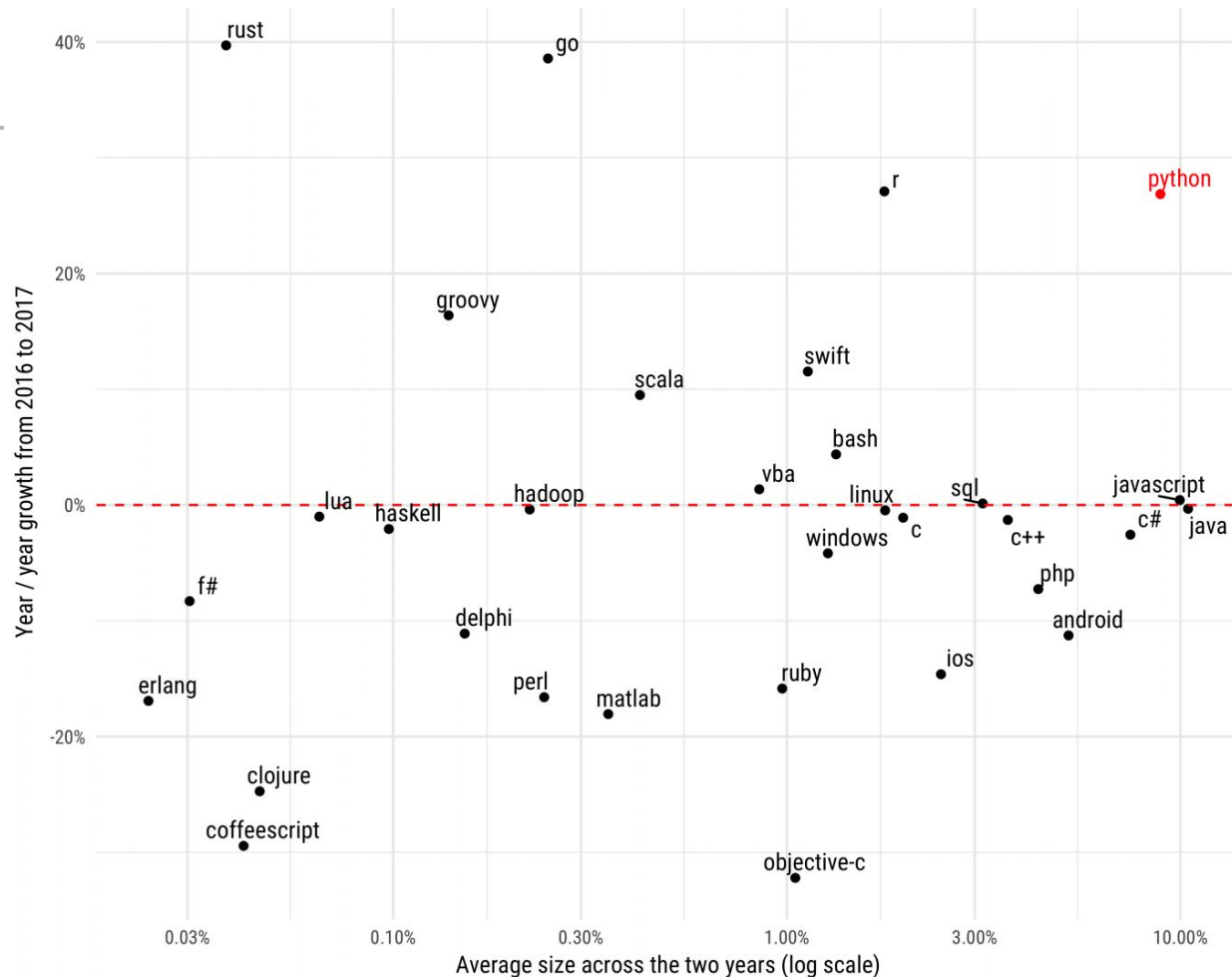
Why Python?

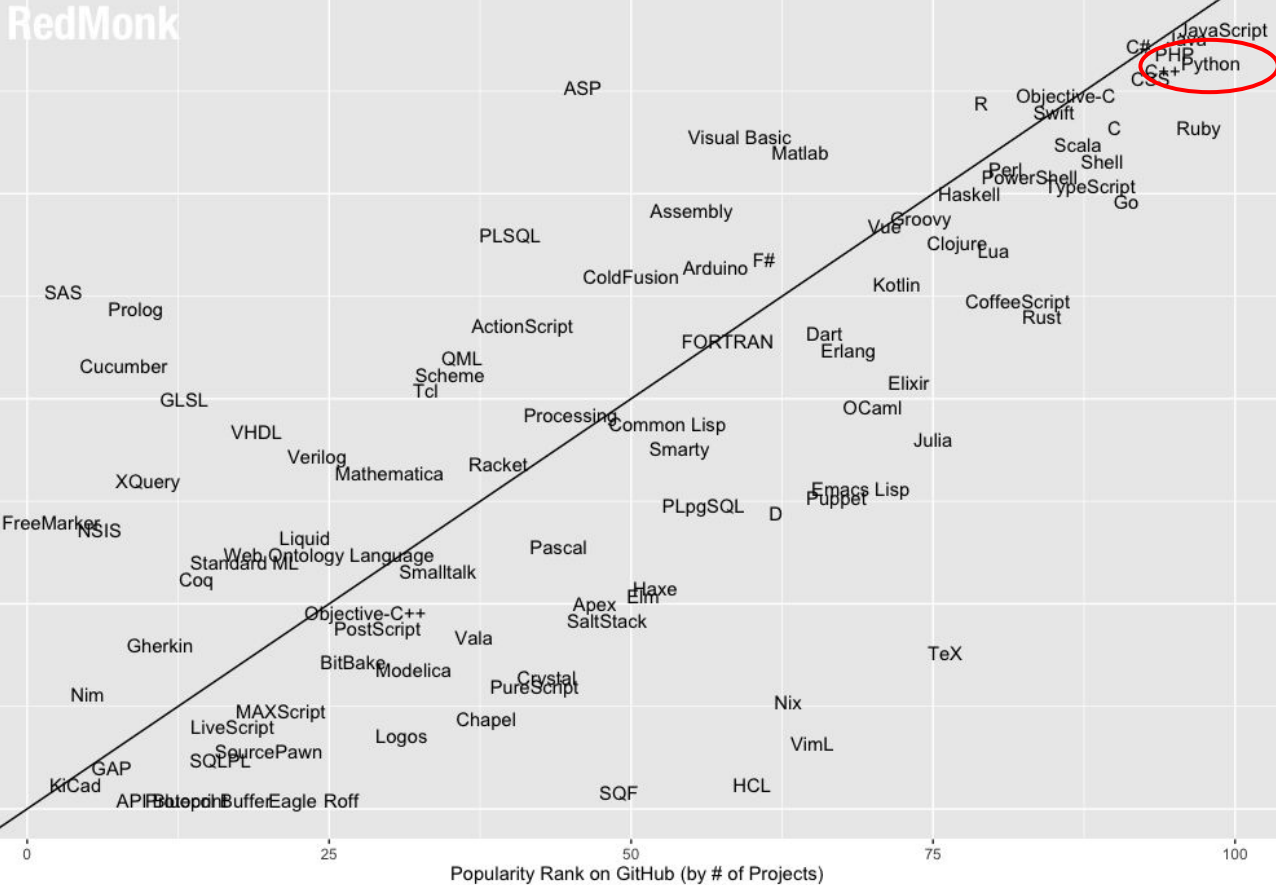
Projections of future traffic for major programming languages

Future traffic is predicted with an STL model, along with an 80% prediction interval.



Why Python?





Source: <https://redmonk.com/sogrady/2018/08/10/language-rankings-6-18/>

Learn to Learn

What's popular today may not be popular tomorrow

- Strategy #1: learn *concepts* rather than recipes.

Classifiers

What is a classification problem?

What are the desirable properties for a classifier? How does this vary for different applications?

What are typical things to watch out (what can fail?) when solving a classification problem?

VS

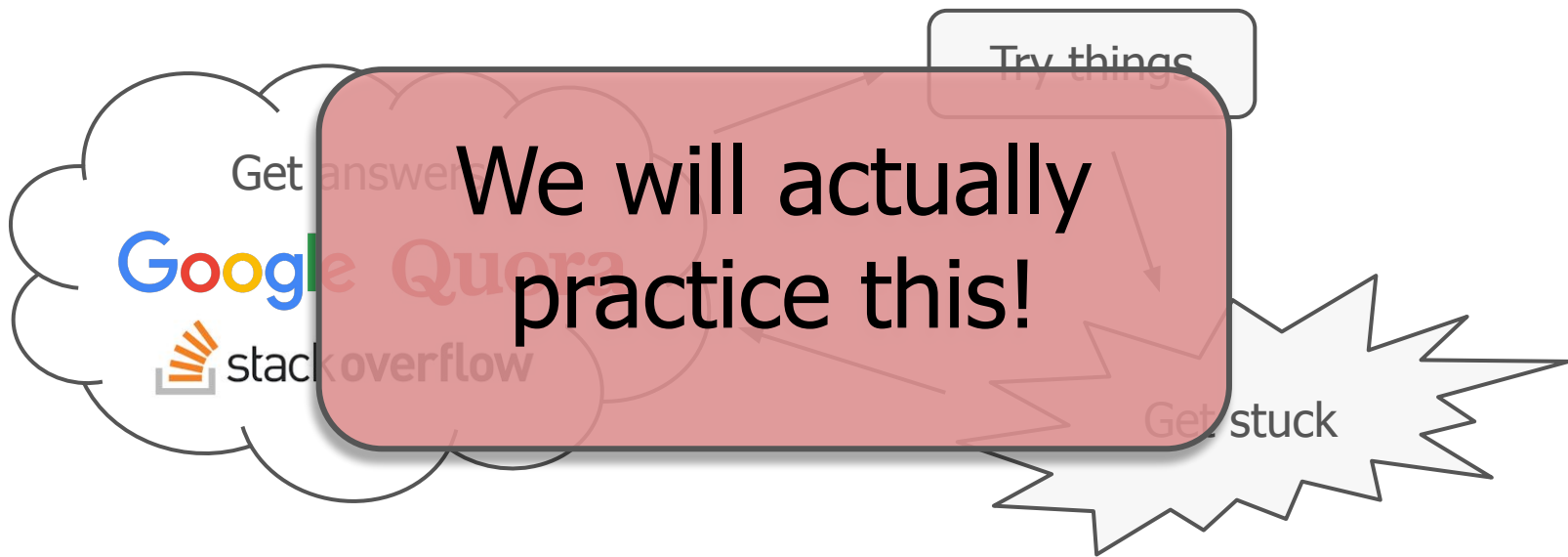
Classifiers

linear least squares, naive Bayes, logistic regression, support vector machines, linear discriminant analysis, k-nearest neighbor, decision trees, boosted trees, random forests, perceptron, neural networks, convolutional nets, deep nets,...

Learn to Learn

What's popular today may not be popular tomorrow

- Strategy #2: learn how to figure things out!



Administrative stuff

- Syllabus (see Canvas page)