ECE 20875

Python for Data Science

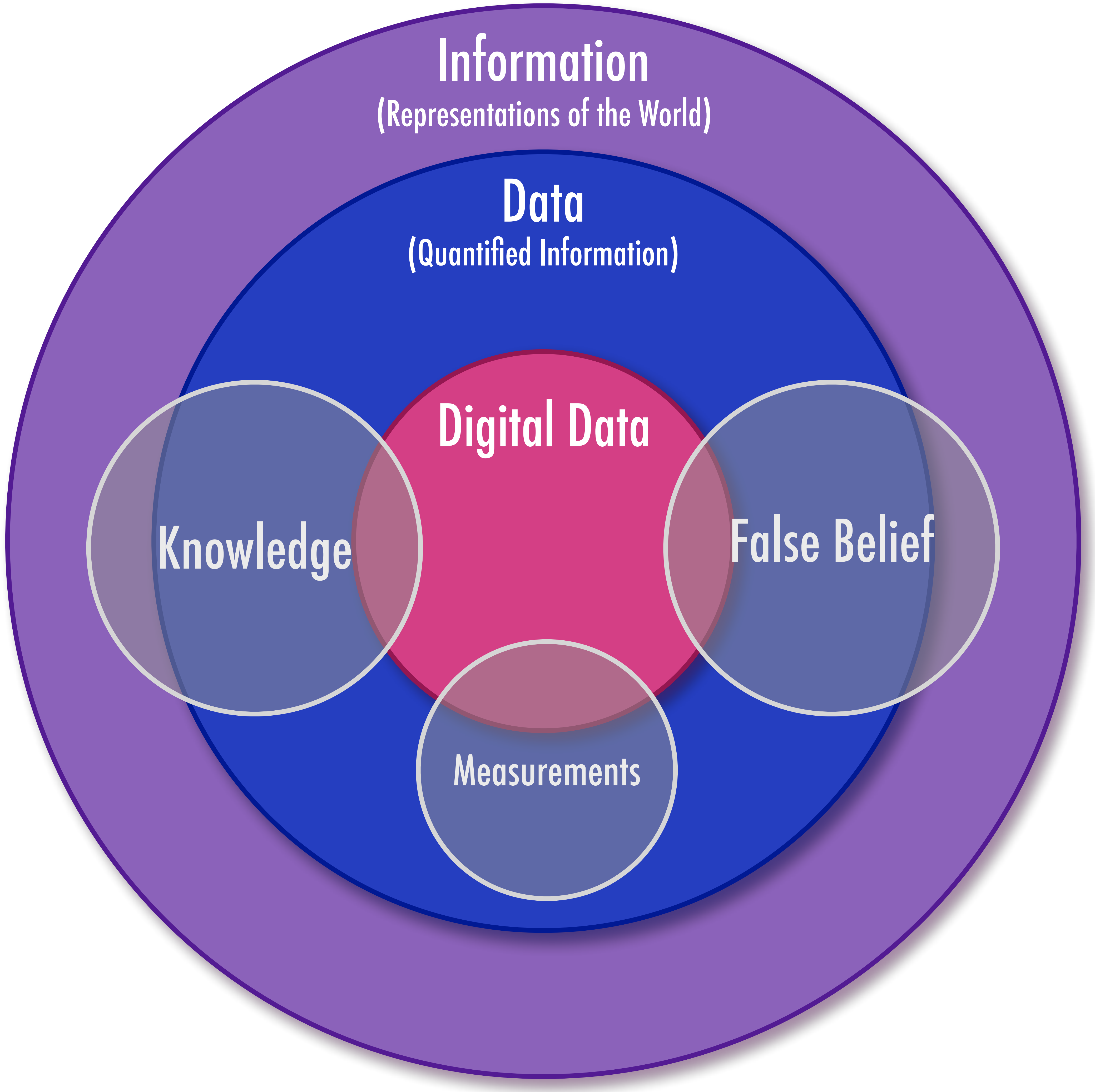
David Inouye and Qiang Qiu

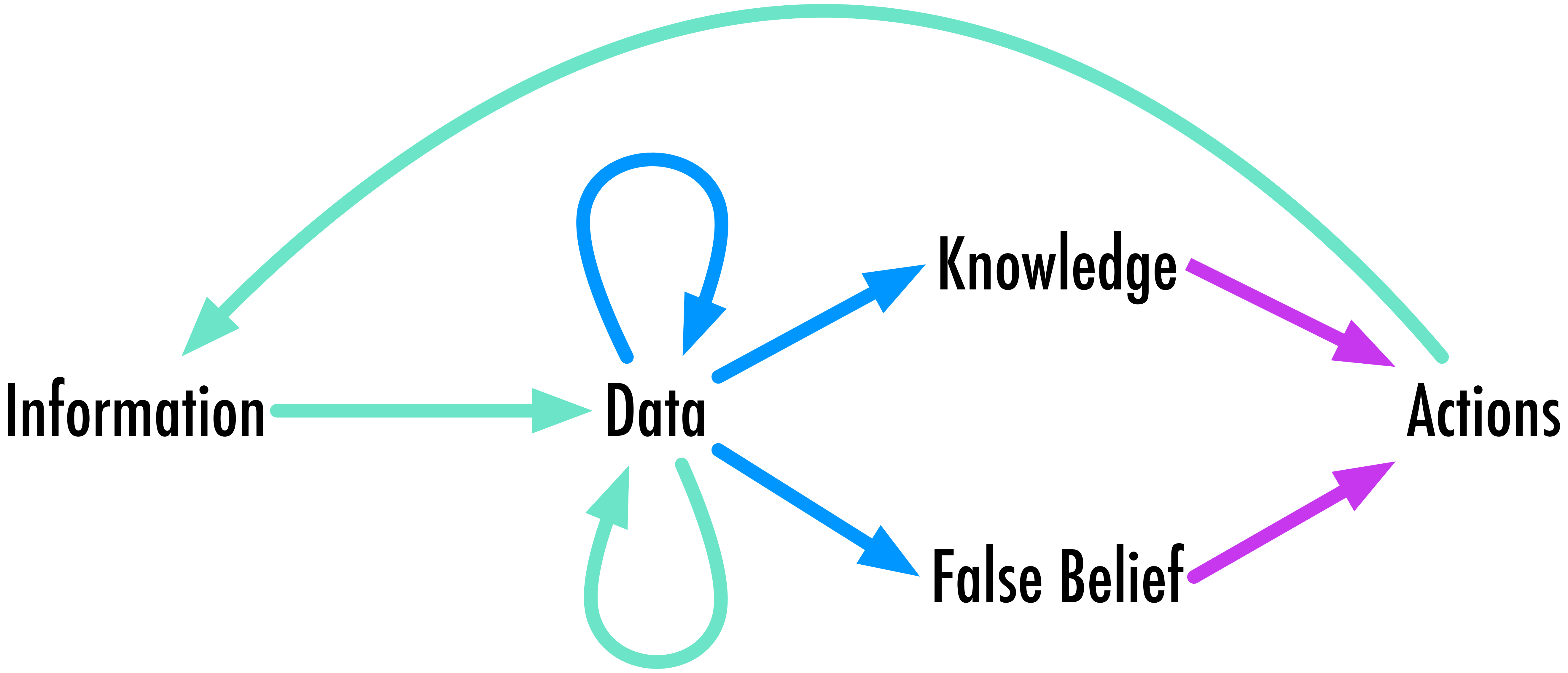
(Adapted from material developed by Profs. Milind Kulkarni, Stanley Chan, Chris Brinton, David Inouye)

Section I: EE 129 + Zoom (Inouye)
Section II: ME 1061 + Zoom (Qiu)
Online Section: Zoom (Inouye)

what is data?

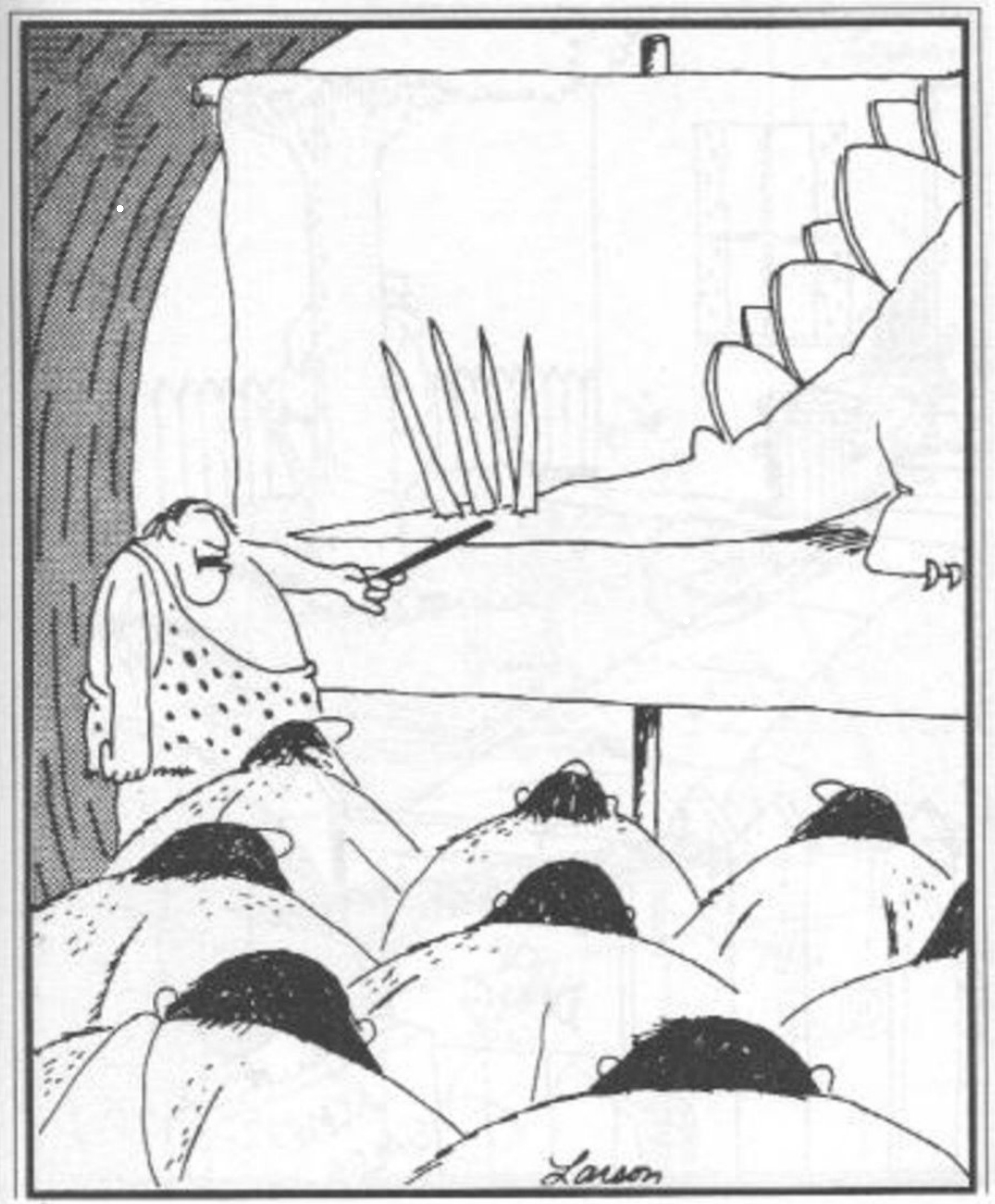
lots of different definitions





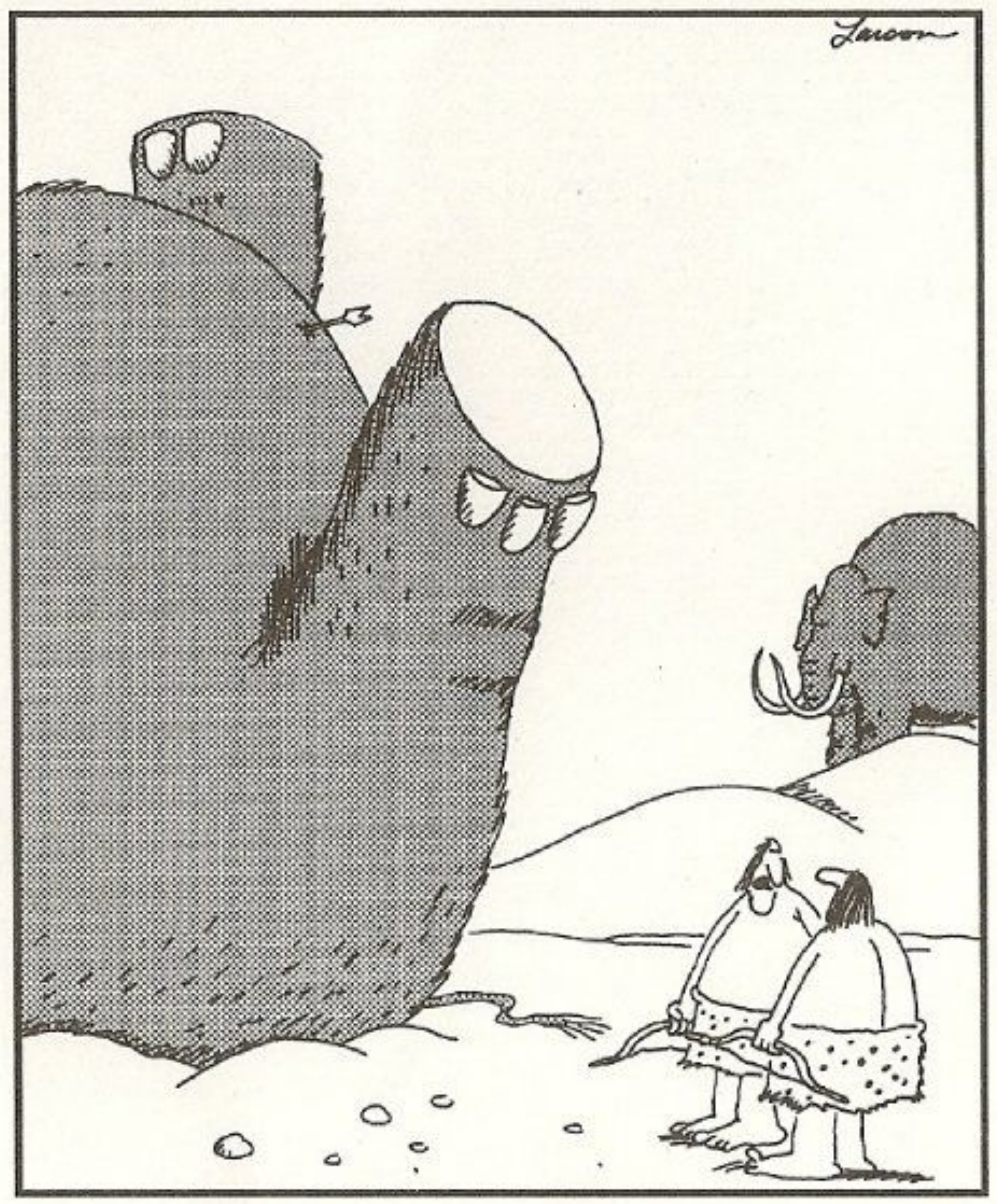
humans have used data forever

- Ever since Thag Simmons first thought, "Last time, we only sent two people to hunt the smilodon. Maybe this time we should send three?"



"Now this end is called the thagomizer . . . after the late Thag Simmons."

why do we use data?

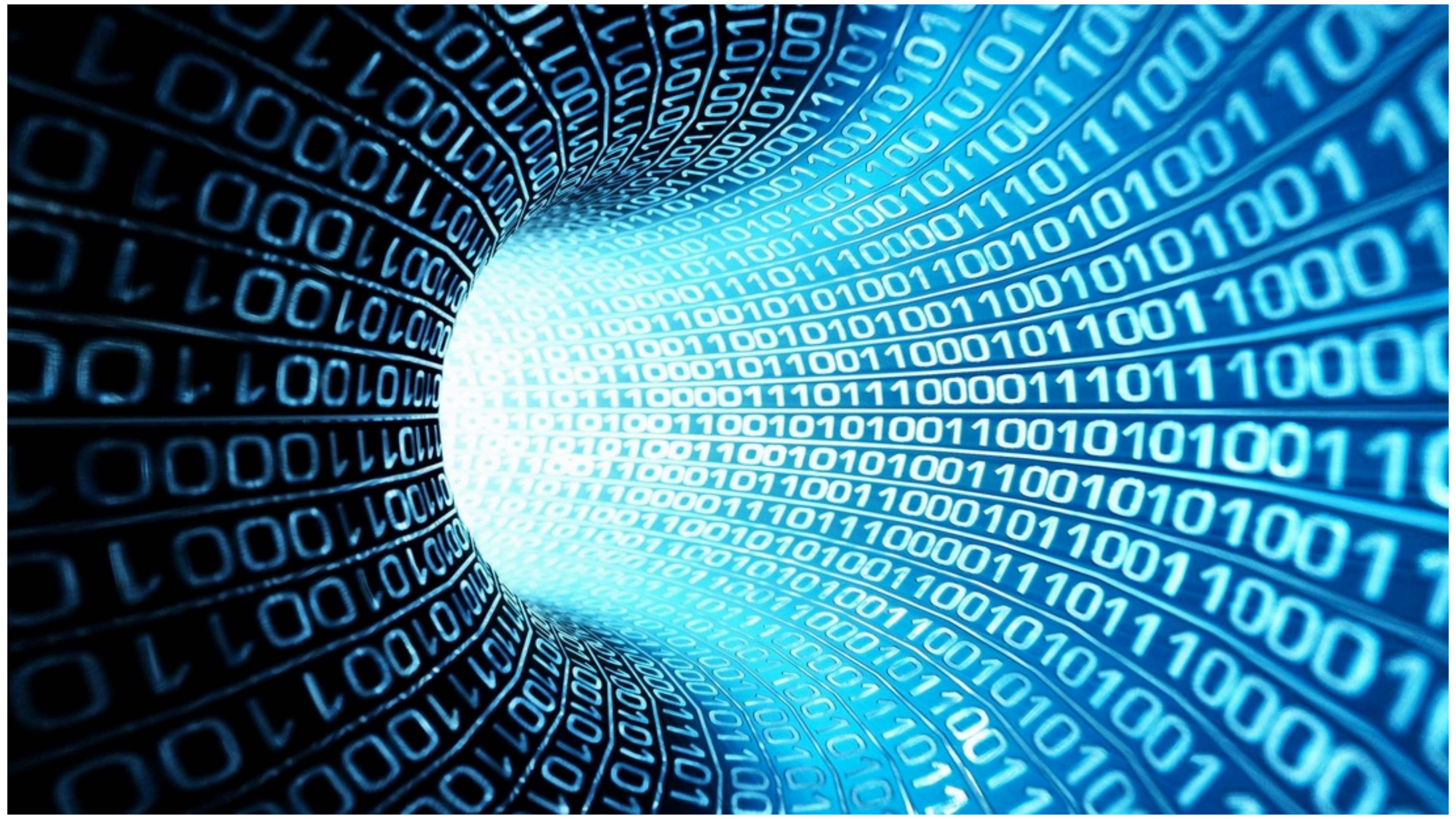


"We should write that spot down."

- Analyzing data helps us make decisions and take actions

what has changed?

- There's a lot more data
- Machines can also collect (and in turn use) it
- And we're trying to do more with it



a parable of purdue professors

Prof. Seungyoon Lee (Comm)
analyzes social media behavior
to understand how social networks
help people process information

Prof. Bryan Pijanows
sound recordings fro
ecological change

Are they doing data science?

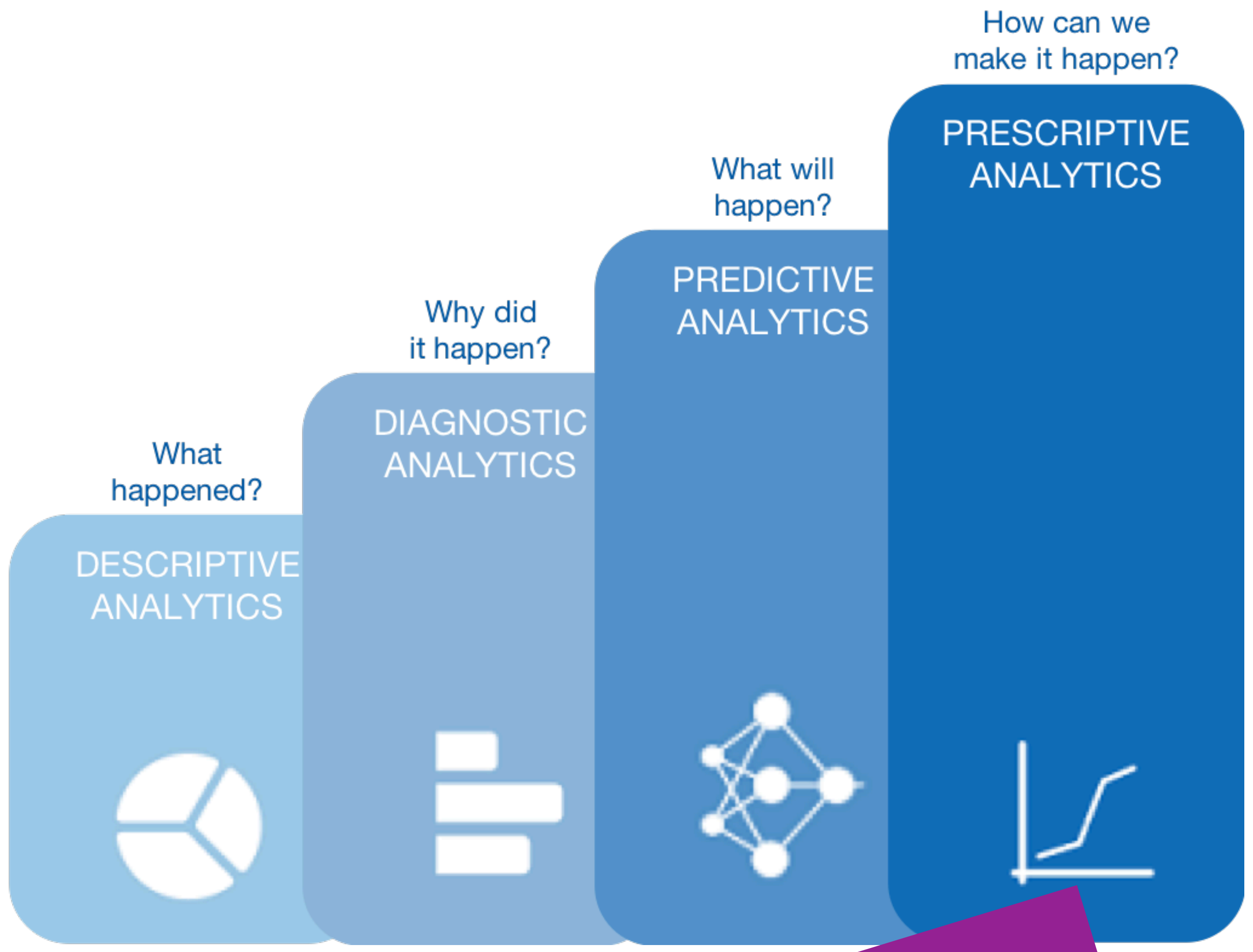
Neville (CS) builds
earning tools
s and networks

Prof. Milind Kulkarni (ECE) builds systems
to make data analyses run faster

Prof. Chris Brinton (ECE) develops
algorithms for modeling and
optimizing social and
communication networks from data

what is data science?

- Collecting data from a wide variety of sources and putting them into a consistent format?
- Making observations about patterns in data?
- Visualizing trends in data?
- Identifying similarities between data points?
- Making predictions about what will happen in the future?
- Prescribing courses of action to take based on forecasts?
- Developing new machine learning and data mining algorithms?
- Accelerating analysis algorithms?



Yes!

data science is a lot of things

making predictions from data

identifying patterns in data

building systems for data analysis

dealing with privacy concerns

visualizing data

collecting/organizing data

interpreting data

analyzing data

ethics

writing data analyses

data science is a lot of things

making predictions from data

identifying patterns in data

building systems for data analysis

dealing with privacy concerns

visualizing data

collecting/organizing data

interpreting data

analyzing data

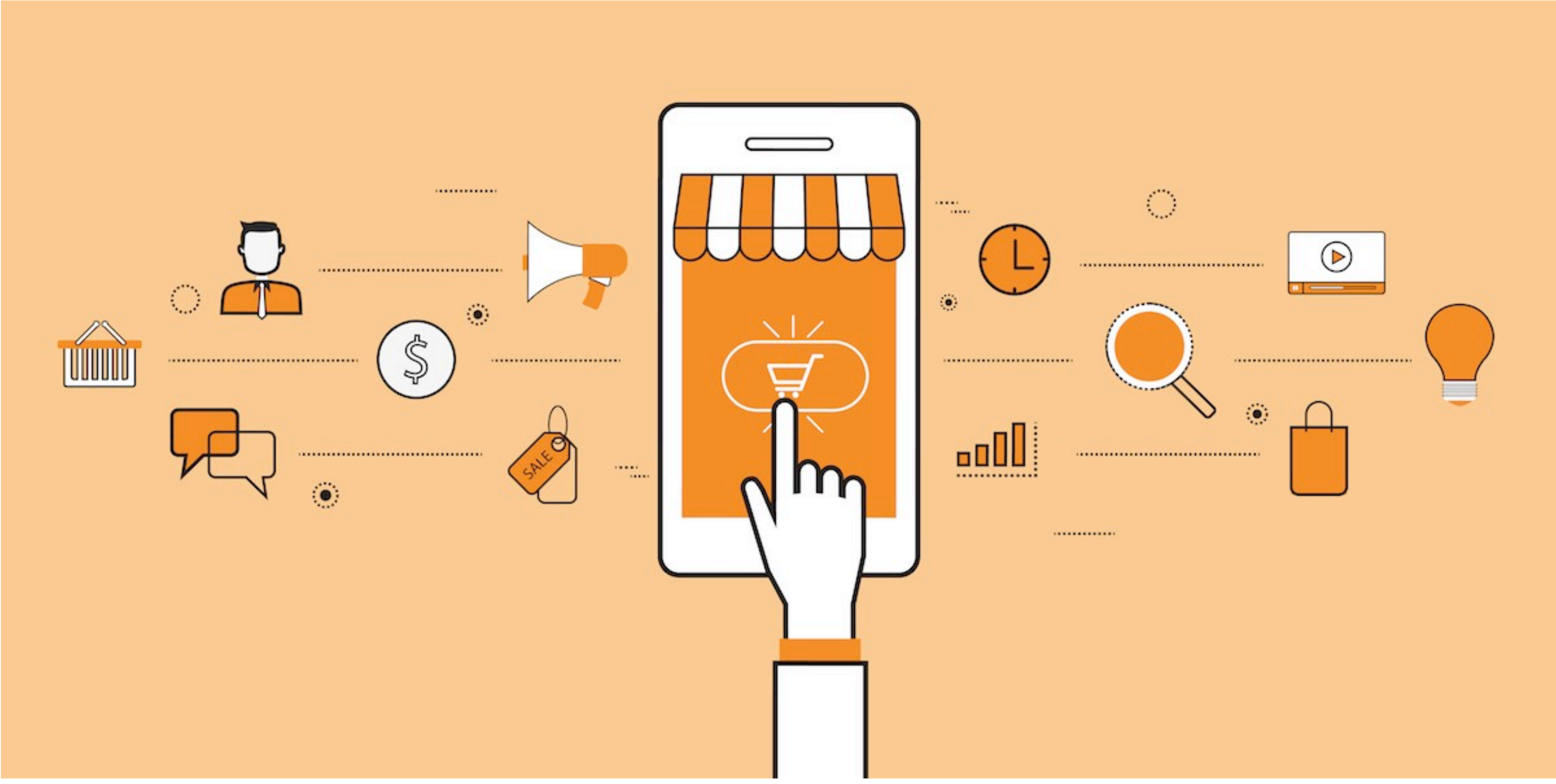
ethics

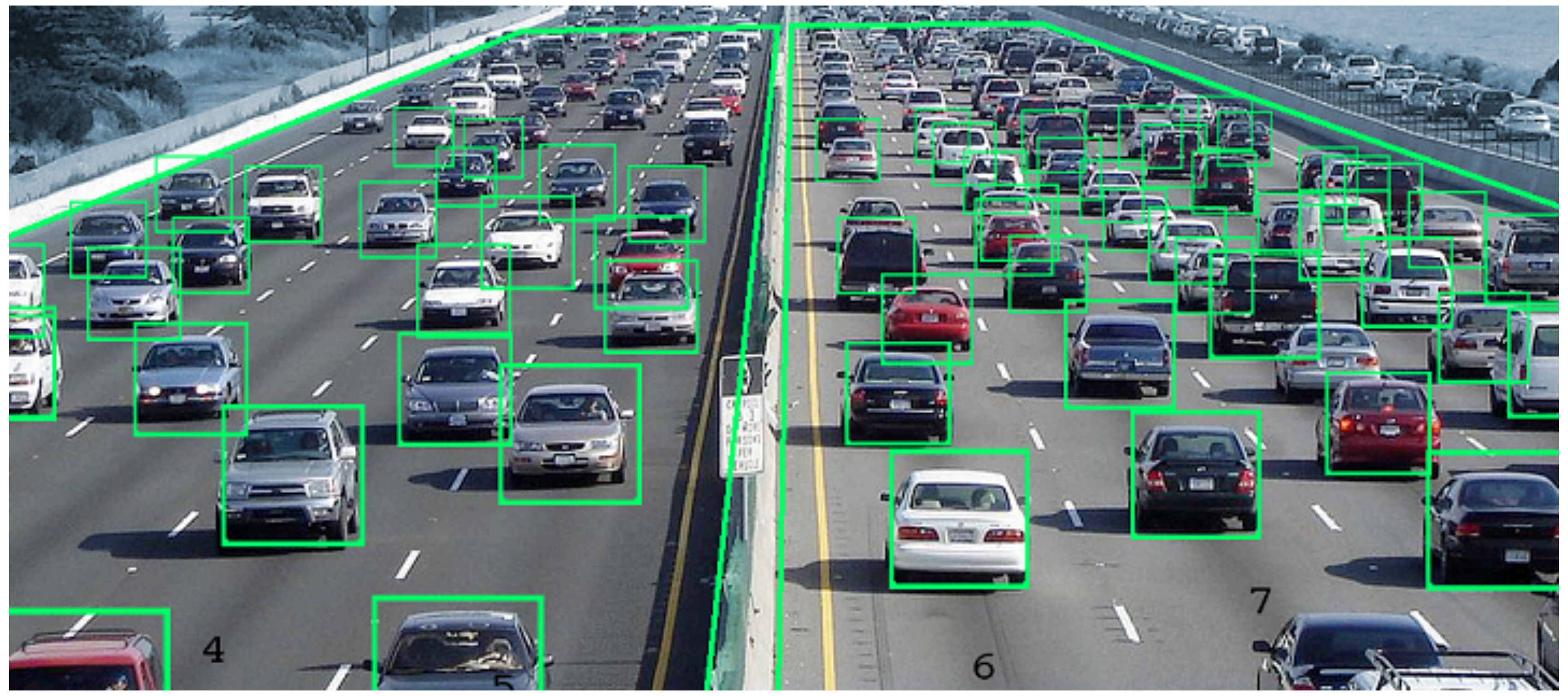
writing data analyses

what industries has it impacted?

- Hard to think of one that is *not* being impacted by data science!

- Medicine: Analytics from wearable trackers, studying disease patterns, …
- Retail: Analyzing consumer behavior, predicting customer satisfaction, …
- Transportation: Assisted/autonomous navigation, predicting equipment failures, …
- Education: Tracking student engagement, personalizing learning content, …





what about python?

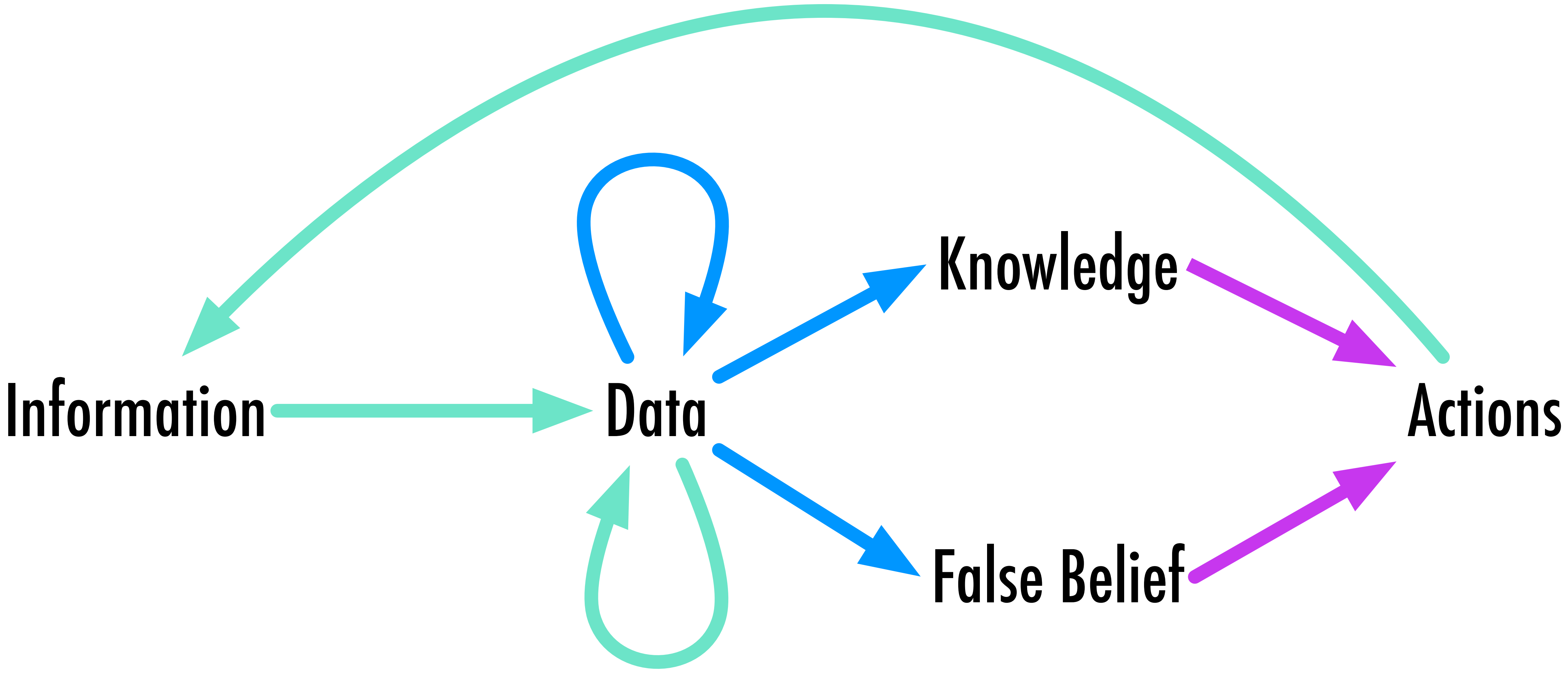
- General purpose programming language, first appeared in the 90s
- Easily recognized by use of whitespace indentation rather than { } brackets to enhance readability
- Becoming the industry standard for data science (competing with R)
- Many useful, open-source libraries: numpy, pandas, matplotlib, pytorch
- And standard control functions (e.g., loops) from lower-level languages to help structure programs

```
# Build the TensorFlow graph.
g = tf.Graph()
with g.as_default():
  # Build the model.
  model = show_and_tell_model.ShowAndTellModel(
      model_config, mode="train", train_inception=FLAGS.train_inception)
  model.build()

  # Set up the learning rate.
  learning_rate_decay_fn = None
  if FLAGS.train_inception:
    learning_rate = tf.constant(training_config.train_inception_learning_rate)
  else:
    learning_rate = tf.constant(training_config.initial_learning_rate)
    if training_config.learning_rate_decay_factor > 0:
      num_batches_per_epoch = (training_config.num_examples_per_epoch /
                               model_config.batch_size)
      decay_steps = int(num_batches_per_epoch *
                        training_config.num_epochs_per_decay)
```

landscape

- This is an introductory programming course that emphasizes data science problems with some math
- Other data science courses in ECE:
  - ECE 30010 - Introduction to Machine Learning and Pattern Recognition
  - ECE 47300 - Introduction to Artificial Intelligence
  - ECE 57000 - Artificial Intelligence
  - ECE 59500 - Machine Learning I
- But data science is a Purdue-wide initiative!



syllabus break!

some data analysis examples

data analysis in "practice"

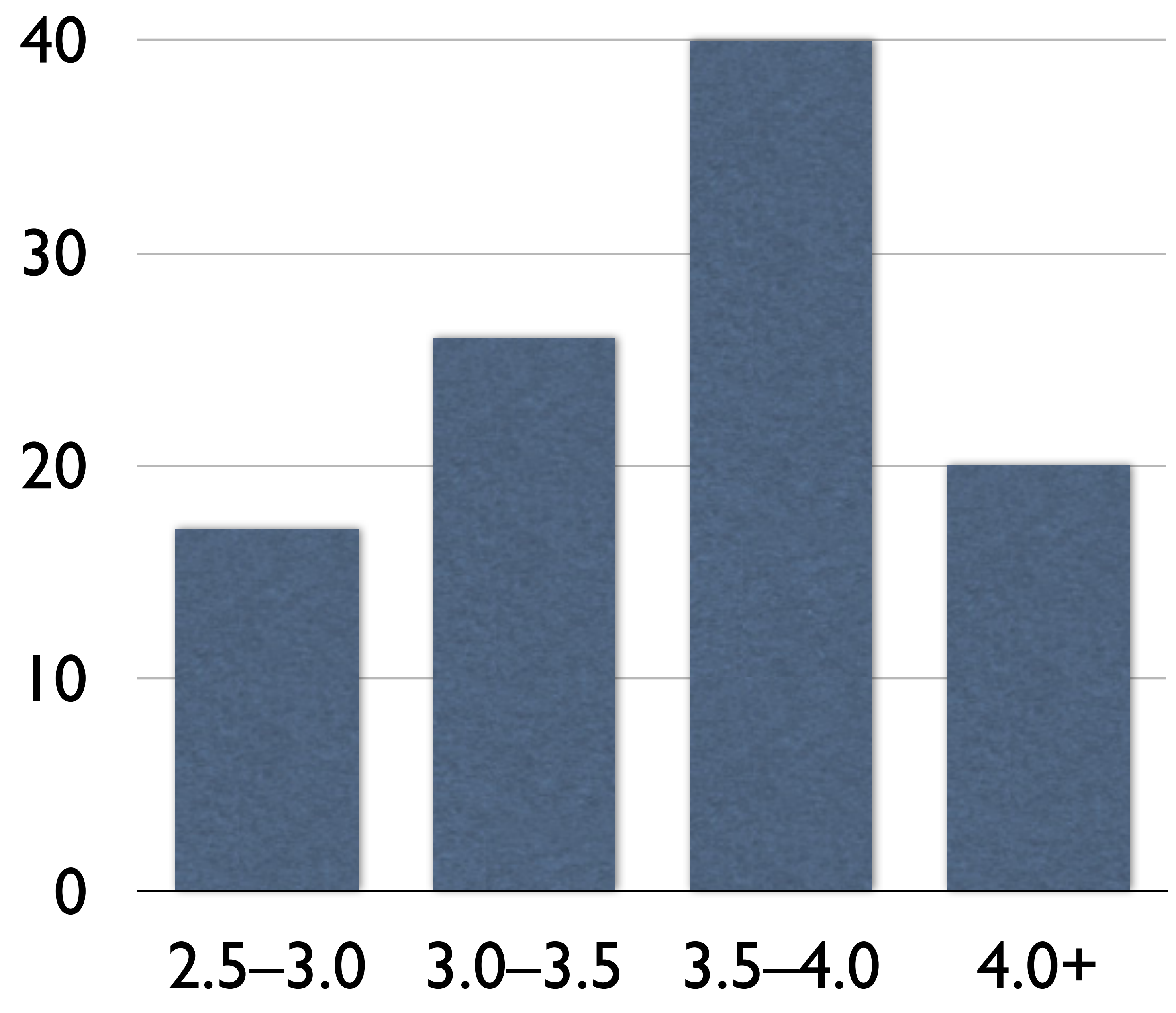
- Lets say we have a data set of applicants to Purdue

Name	High school GPA	SAT Math	SAT R/W	Residence
Jane Doe	4.7	760	700	Indiana
Purdue Pete	3.5	680	620	Indiana
B. O. ller	3.0	800	650	Michigan
Engy Neer	4.2	750	590	North Carolina
Mark Faller	3.8	780	550	New Jersey
…	…	…	…	…

- What might we want to learn about them?

descriptive statistics

- Which students come from which states?
- What is the distribution of GPAs? SAT scores?
    - GPAs may need to be *normalized* to a consistent range across all schools
- Can build *histograms*, e.g., for the GPAs
    - But how do we know how big to make the buckets?

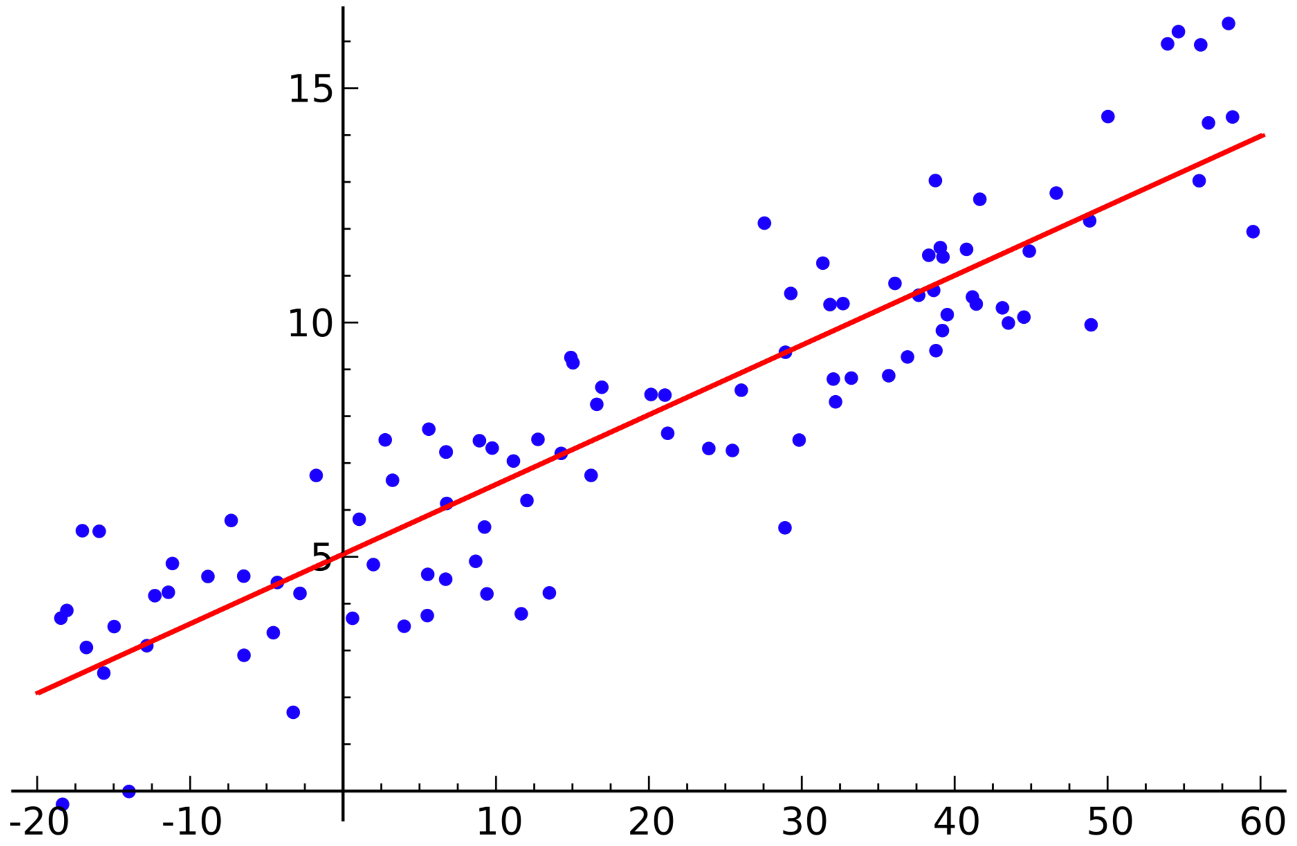


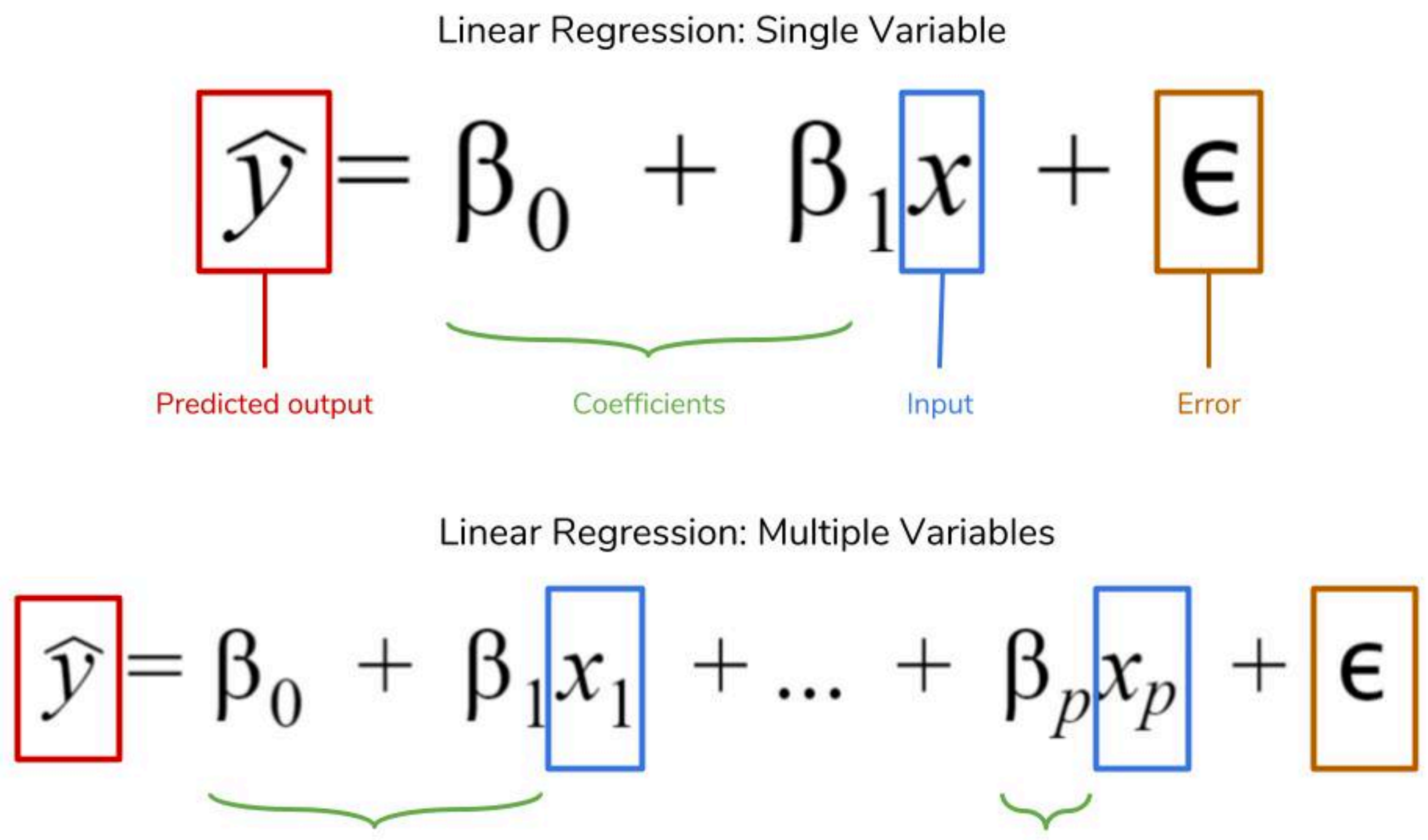
reasoning about data

- How do Purdue applicants compare to the national average?
  - *Mean* GPA of applicants: 3.6
- Is this high or low?
  - Can *sample* GPA of all high school students
- Suppose we collect 1000 GPAs and find a mean of 3.4
  - Does this mean Purdue students have a higher GPA on average?
- Need more information! In particular …
  - Was the sampling method we used *unbiased*?
  - What is the *variance* of the sample collected (i.e., the spread of GPAs)?
  - What *confidence interval* can be built for the population mean (i.e., what is the likely range of the true mean GPA)?

making predictions

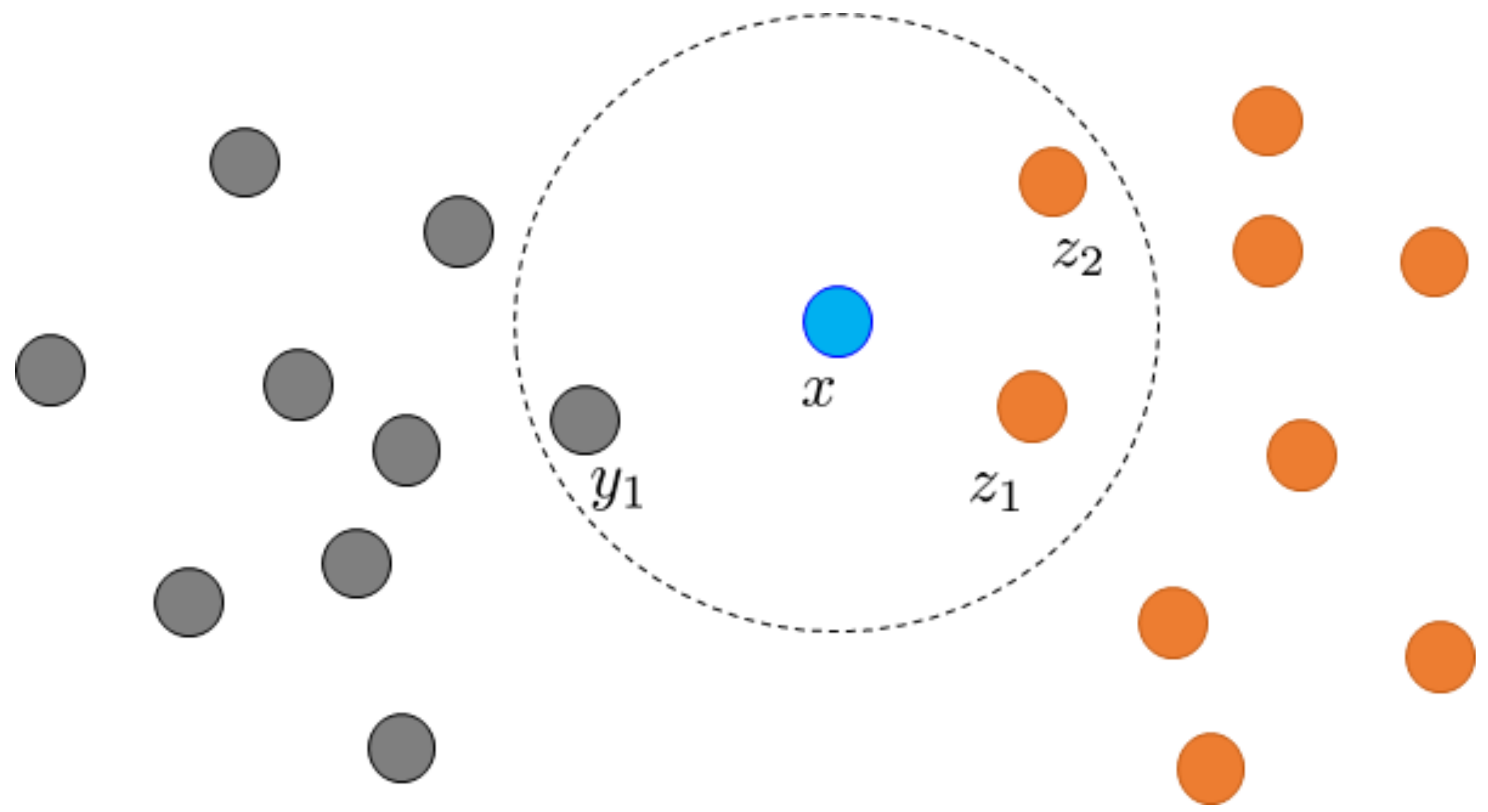
- Can we predict how successful a particular applicant might be at Purdue?
    - How do we define success? GPA?
- Idea: Look at the application statistics of the *current seniors* and see if there is a relationship between these statistics and their current GPA
- One way to find a relationship is using *linear regression*
    - Might tell you something like: "a Purdue student's GPA can be predicted mostly by their high school GPA, with their SAT score having a lighter influence"
- Many other prediction algorithms exist too





classification

- Can we make admissions decisions quicker through automation?
- Idea: Compare each applicant's statistics to past applicants that were admitted, and to those that were rejected
- Train a *classifier* to analyze these past applicants and maximize the ability to predict whether a student would be accepted or not
  - For example, a *k-nearest neighbor* classifier would assess whether a given applicant is more similar to the pool of admitted applicants or to the rejected applicants
  - Why might we run into trouble here?



clustering

- What if we want to identify groups of students beyond “admitted” vs. “rejected”?
- Idea: See if students cluster together according to some measure of *distance*
  - Some students look more like “nearby” students than students that are “far away”
- Important question: What *features* of students should be considered for the clustering?
  - E.g., maybe don’t consider something like hair color!
- With *k-means clustering*, *k* groups of students would be extracted based on “closeness”