I remember PollEverywhere

A) Yes
B) No
“Big data will become a key basis of competition, underpinning new waves of productivity growth, innovation, and consumer surplus.”

– McKinsey
Big Data is ...

- Volume
- Velocity
- Variety
- Veracity
2012 IPO: $3.3 billion

2017: $8.5 billion
Big Data

- Big $n$ vs. big $p$
- How is big data different?
  - use available large-scale data rather than annotating data
  - heterogeneous ("variety")
- Semi-parametric or non-parametric methods
Different methods work best at scale

- Confusion set disambiguation
  - Choose the correct word in the set given the context
    - {principle, principal}
    - {then, than}
    - {to, two, too}
    - {weather, whether}

Figure 1. Learning Curves for Confusion Set Disambiguation
The unreasonable effectiveness of data

- Scene completion using millions of photographs
How to handle big data?

- Dimensionality reduction
- Sampling
- Streaming
- Hadoop/MapReduce
Different data handling:
- Mostly unstructured data objects (Schema-less NoSQL)
- Many attributes and data sources
- Data sources added and/or updated frequently
- Quality is unknown

Different programming philosophy:
- Distributed, fault tolerant programming
What is the slowest part of big data analysis?

A) Multiplying $X'X$
B) Inverting a matrix $(X'X)^{-1}$?
C) Reading $X$ from disk to memory?
D) Other?
Synchronous Variant

Model Parallelism
Data Parallelism

References
http://developer.yahoo.com/hadoop/
http://code.google.com/edu/parallel/mapreduce-tutorial.html
Data is divided across multiple machines ("mappers")

Each mapper does the same thing to different data

Results are combined ("reduced")

How easy is it to do in map-reduce?

- Linear regression
- Linear regression with feature selection
- SVM
- k-NN
- K-means / EM

A) Easy
B) Hard
C) Impossible
Good tools

- LDA
  - Mallet
  - Factorie
- Deep Nets
  - Theano
  - Caffe, Torch
  - Tensorflow
MapReduce: Hadoop’s Original Data Processing Engine

Key Advances by MapReduce:

- **Data Locality**: Automatic split computation and launch of mappers appropriately.
- **Fault-Tolerance**: Write out of intermediate results and restartable mappers meant ability to run on commodity hardware.
- **Linear Scalability**: Combination of locality + programming model that forces developers to write generally scalable solutions to problems.

Credit: cloudera
In Hadoop

- **Hive**
  - data warehouse: data summarization, query, and analysis.

- **Pig, Crunch**
  - high-level platform for creating MapReduce programs

- **Mahout**
  - scalable machine learning and data mining

- **Solr**
  - enterprise search platform built on Apache Lucene

- **Hue**
  - visualization
Spark

- Combines SQL, streaming, and complex analytics
- Often runs on Hadoop
  - or Mesos, or standalone, or in the cloud
- Bindings to
  - Java, Scala, Python, R, NLTK ...
- MLlib Machine Learning Library
  - Faster than Mahout

Seems to be replacing Hadoop
Increasingly use a “deep stack”

BDAS Stack

Cancer Genomics, Energy Debugging, Smart Buildings

BlinkDB  Sample Clean  MLBase  SparkR
Spark Streaming  SparkSQL  GraphX  MLlib

Apache Spark  Velox Model Serving

Tachyon

HDFS, S3,

Apache Mesos  Yarn
Increasing in the cloud

◆ X as a Service
  ● SaaS (software)
  ● PaaS (platform)
  ● IaaS (infrastructure)

◆ It’s easy to spin these up on AWS or MS Azure …

http://www.mazikglobal.com/
**Tools are changing rapidly**

- **Currently hot:**
  - **SMACK:** Spark, Mesos, Akka, Cassandra and Kafka
    - **Spark** – fast engine for distributed large-scale data processing
    - **Mesos** - distributed systems kernel
    - **Akka** - toolkit and runtime for building highly concurrent, distributed, and resilient message-driven applications
    - **Cassandra** – distributed database
    - **Kafka** - distributed publish-subscribe messaging system
  - **Tensorflow**

*But the fundamentals we learned in this class are not changing!*
Speeding up your ML code

Lyle Ungar

Photo credit http://allinguide.com/best-tips-how-to-speed-up-your-wordpress-website-or-blog/
Your ML code runs too slow; What can you do?
How to speed up your ML?

- Speed up the code
  - Use a faster language
  - Use a cluster/multicore machine /GPU
  - Vectorize

- Use a streaming algorithm
  - In features or observations

- Develop on a subset of the data
  - Or a subset of the features (univariate preprocessing)

- Do dimensionality reduction
How to speed up your ML?

◆ Pick a faster algorithm
  ● Logistic regression ➔ ?
  ● Kernelized SVM ➔ ?
  ● Stepwise regression ➔ ?
  ● K-NN ➔ ?
Pick a faster algorithm

- Logistic regression → linear regression
- Kernelized SVM → linear SVM
- Stepwise regression → stagewise regression
- K-NN → K-means
How to speed up your ML: True/False

- Sparse code runs faster?
- Vector-based code runs faster?
- Models based on principle components are usually faster than one in the original features?
Take-Aways

- Data variety complicates machine learning
  - Data wrangling, complex regularization

- Many ways to speed up code
  - Vectorize, run on GPU
  - Use online algorithms
  - Use data-parallel methods (map-reduce)

- Lots of good software
  - SKLearn, spark, tensorflow