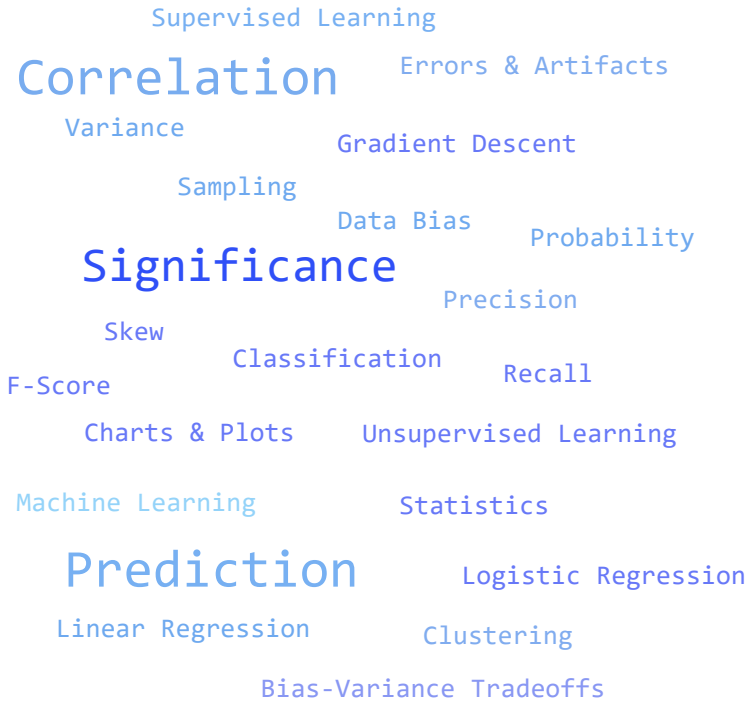


Data Science 1:
Introduction to Data Science

Data Science in the Wild

Winter 2025

Wolfram Wingerath, Jannik Schröder

Department for Computing Science
Data Science / Information Systems

Lecture slides based on content from “The Data Science Design Manual” (Steven Skiena, 2017) and associated course materials generously made available online by the author at https://www3.cs.stonybrook.edu/~skiena/data-manual/.

Special thanks to Professor Skiena for sharing these valuable teaching resources!

Semester Schedule

CW 42	14. Oct	Lecture	1	Orga & Intro	1-26
CW 43	21. / 23. Oct	Lecture + Exercises	2	Probability, Statistics & Correlation	27-56
CW 44	28. Oct	Lecture	3	Data Munging, Cleaning & Bias	57-94 / “Invisible Women”
CW 45	04. / 06. Nov	Lecture + Exercises	4	Scores & Rankings	95-120
CW 46	11. Nov	Lecture	5	Statistical Distributions & Significance	121-154
CW 47	18. / 20. Nov	Lecture + Exercises	6	Building & Evaluating Models	201-236
CW 48	25. Nov	Guest Lecture	7	Data Visualization	155-200
CW 49	02. / 04. Dec	Lecture + Exercises	8	Intro to Machine Learning	351-390
CW 50	09. Dec	Lecture	9	Linear Algebra	237-266
CW 51	16. / 18. Dec	Lecture + Exercises	10	Linear Regression & Gradient Descent	267-288
CW 02	06. Jan	Lecture	11	Logistic Regression & Classification	289-302
CW 03	13. / 15. Jan	Lecture + Exercises	12	Nearest Neighbor Methods & Clustering	303-350
CW 04	20. Jan	Lecture	13	Data Science in the Wild	391-426
CW 05	27. / 29. Jan	Lecture + Exercises	14	Q&A / Feedback	
CW 06	03. / 04. Feb	Oral Exams (Block 1)	Preparation in our last session („Oral Exam Briefing“)		
CW 13	24. / 25. Mar	Oral Exams (Block 2)			

Data Science and Big Data

The buzzword “Big Data” presumes analysis of truly massive data sets:

- think: Twitter, Facebook, Amazon, Google…
- all images on Flickr
- genome sequences of thousands of people
- Web logs for major websites

Working with data generally gets harder with size

How Big Is…

- Twitter (600 million tweets/day)
- Facebook (>600 TB incoming data per day)
- Google (3.5 billion search queries/day)
- Instagram (52 million photos per day)
- Apple (130 billion app downloads)
- Email (205 billion message/day)

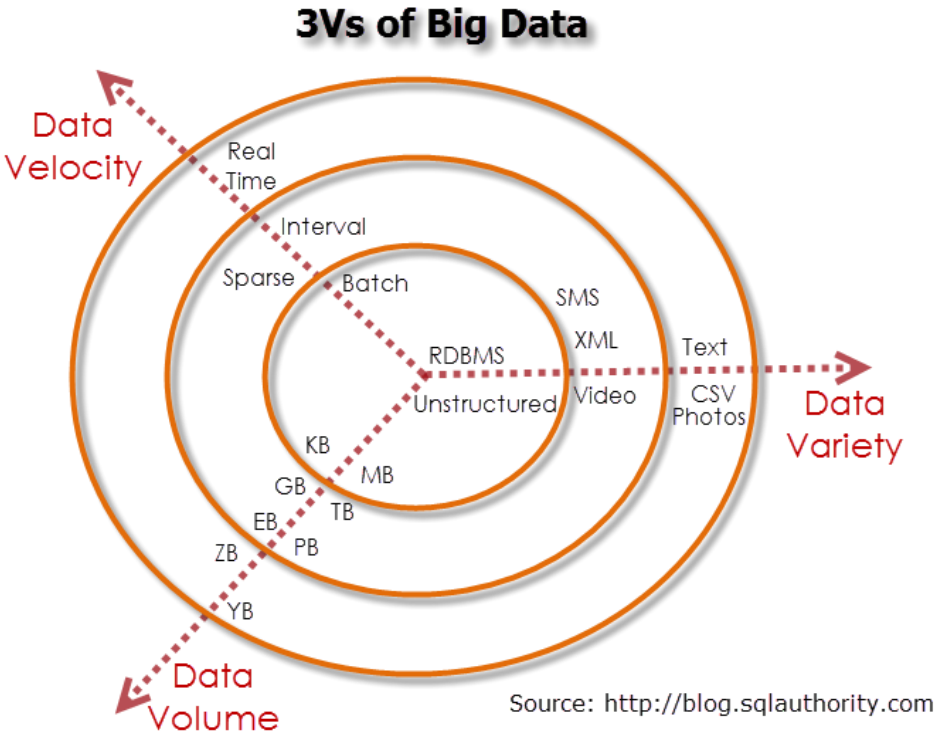
https://www.internetlivestats.com

The Three V's of Big Data

Student projects are typically batch problems on MB scale CSV-type data.

There are more Vs to worry about!

(Veracity, Value, Visualization, …)



Big Data as Bad Data

Massive data sets are the result of opportunity instead of design, with problems of:

- Unrepresentative participation (bias)
- Spam and machine-generated content
- Power-laws mean too much redundancy
- Susceptibility to temporal bias (e.g Google Flu Trends)

Properties of a Good Model

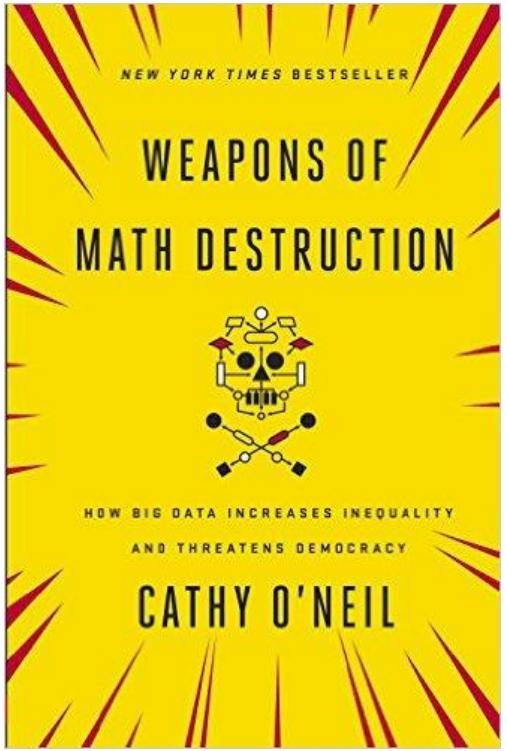
- It uses relevant data, not just data that happens to be available.
- It is transparent, making clear why it is making its decisions.
- There is a clear measure of success, and an embedded feedback mechanism to evaluate and learn from it.

Weapons of Math Destruction

It is important to understand the potential harm data-driven models can cause.

Correlation is not causation, but models so-trained can trigger actions and feedback mechanisms, resulting in self-fulfilling prophecies.

It is important for us as data scientists to think about societal issues in a constructive way.



Large-Scale Machine Learning

The learning algorithms we have studied generally do not scale well to huge data sets.

- Models with few parameters cannot really benefit from large numbers of examples.
- Algorithmic complexity must be near linear to run on large data sets.
- Big matrices better be sparse for big data.

Customization and Specialization

Big data on all your customers translates to modest data on each of many individuals.

Customization means training large numbers of small models, only possible with big data.

Filtering Data

An important benefit of Big Data is that you can discard much of it to make analysis cleaner.

English accounts for only 34% of all tweets on Twitter, but you can exclude the rest and leave enough for meaningful analysis.

Filtering away irrelevant or hard-to-interpret data requires application-specific knowledge.

Subsampling Data

It can pay to subsample good, relevant data:

- Cleanly separate training, testing, and evaluation data.
- Simple, robust models generally have few parameters, making Big Data is overkill.
- Spreadsheet-sized data sets are fast and easy to explore.

Subsampling by Truncation

Taking the first *n* records is reproducible and simple but record order often has meaning:

- temporal biases: only analyze old data.
- lexicographic biases: only analyze the A's, e.g. more Arabic names, fewer Chinese.
- numerical biases: ID numbers can encode meaning, e.g. social security numbers.

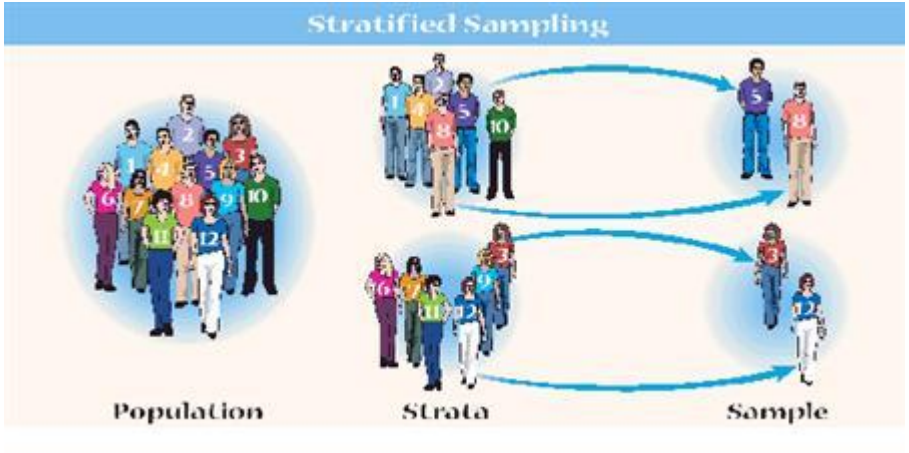
Random Sampling

Randomly sampling records with probability *p* ensures no explicit biases, but:

- Statistical discrepancies ensure some regions will be oversampled.
- Random sampling is not reproducible without the seed and random generator.
- Multiple random samples will not be disjoint.
- Not trivial in some cases (e.g. session data)

Stratified Random Sampling

To ensure no group is statistically over- or under-sampled, we can explicitly sample proportionally from each group.



Uniform Sampling

Sampling records which are congruent to *i mod m* provide a way to balance many concerns:

- Obtain an exact number of records.
- Quick and reproducible.
- Ensures disjoint samples

Twitter uses this method to govern API services (spritzer vs. garden hose vs. fire hose)

Distributed vs. Parallel Processing

The distinction here is how tightly coupled the machines are, roughly:

- **Parallel** processing happens on one machine, through threads and OS processes
- **Distributed** processing happens on many machines, using network communication.

Easy parallel jobs do not communicate much.

Data Parallelism

The easiest way to exploit parallelism partitions big data among multiple machines and trains independent models.

Natural partitions are established by time, clustering algorithms, or given categories.

It is typically hard to combine the results of these runs together later

Grid Search

The easiest way to exploit parallelism involves independent runs on the same data.

Grid search is the quest for the right meta-parameters for training, like deciding the right *k* for k-means clustering.

Multiple independent fits can run in parallel, where in the end we take the best one.

Typical Big Data Problem

- Iterate over a large number of records
- Extract something of interest from each
- Shuffle and sort intermediate results
- Aggregate intermediate results
- Generate final output

Think word counting and k-means clustering

One, Two, Many...

The complexity of distributed processing grows rapidly with the number of machines:

- One: keep the cores of your box busy.
- Two: manually run programs on a few boxes
- Many: employ a system like MapReduce for efficiently managing multiple machines.

Complexities of Scale: Social Gatherings

- 1 person: easy to arrange.
- >2 persons: coordination.
- >10 persons: requires leader in charge.
- >100 persons: requires fixed menu.
- >1000 persons: no one knows many people.
- >10,000 persons: too few hotels for most cities.
- >100,000 persons: someone **might** die that day

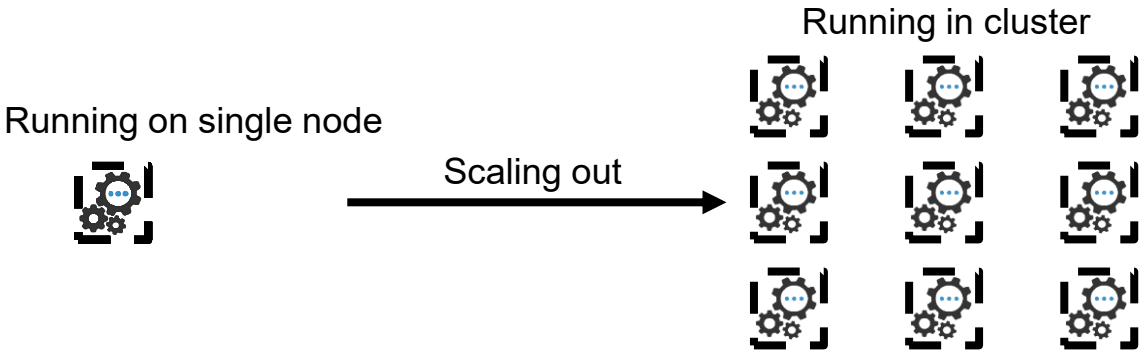
Distribution Challenges

- How do we assign work units to workers?
- What if we have more work units than workers?
- What if workers need to share partial results?
- How do we aggregate partial results?
- How do we know all the workers have finished?
- What if workers die?

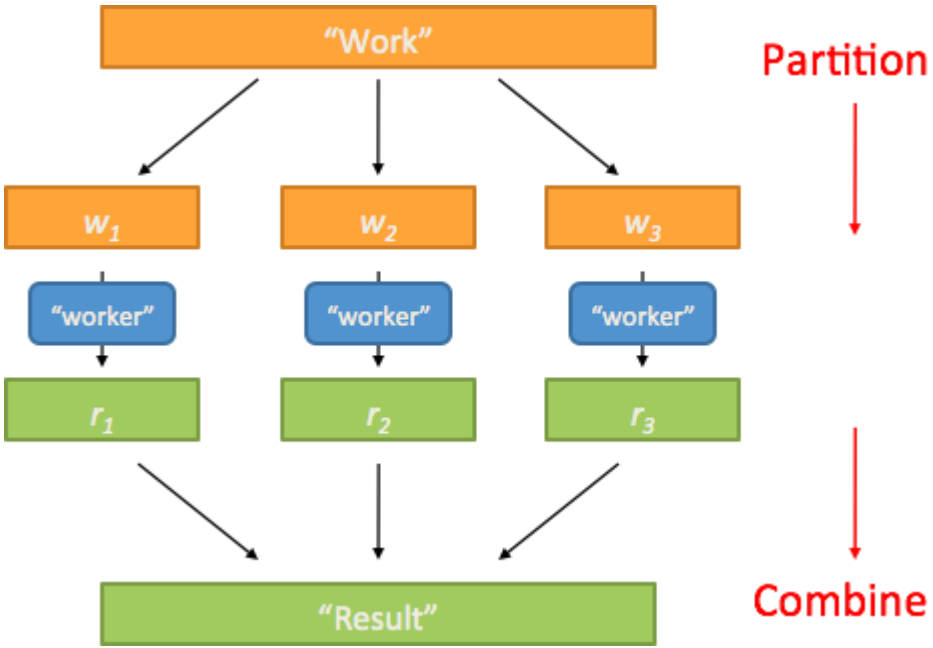
Frameworks: Scale-Out Made Feasible

Data processing frameworks **hide complexities of scaling**, e.g.:

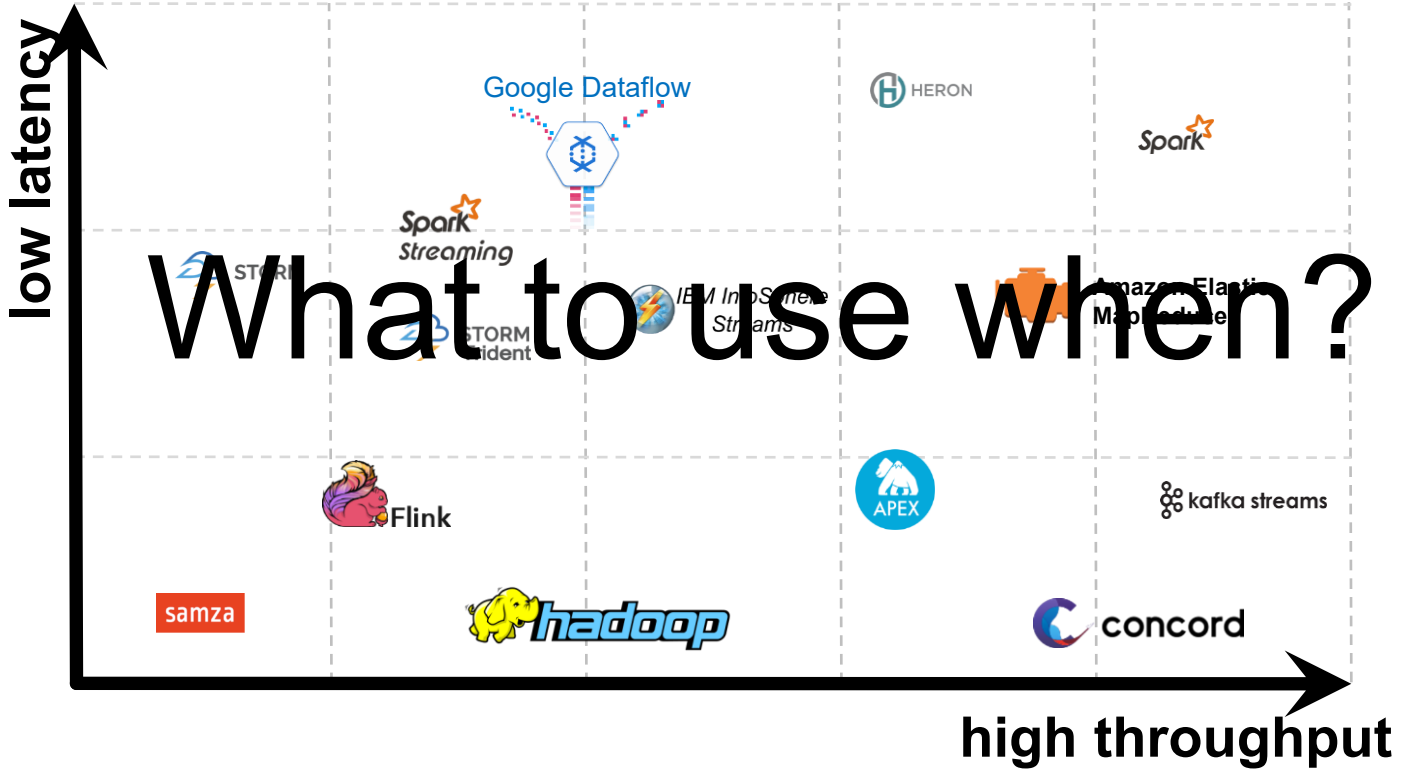
- **Deployment -** code distribution, starting / stopping work
- **Monitoring -** health checks, application stats
- **Scheduling -** assigning work, rebalancing
- **Fault-tolerance** - restarting workers, rescheduling failed work



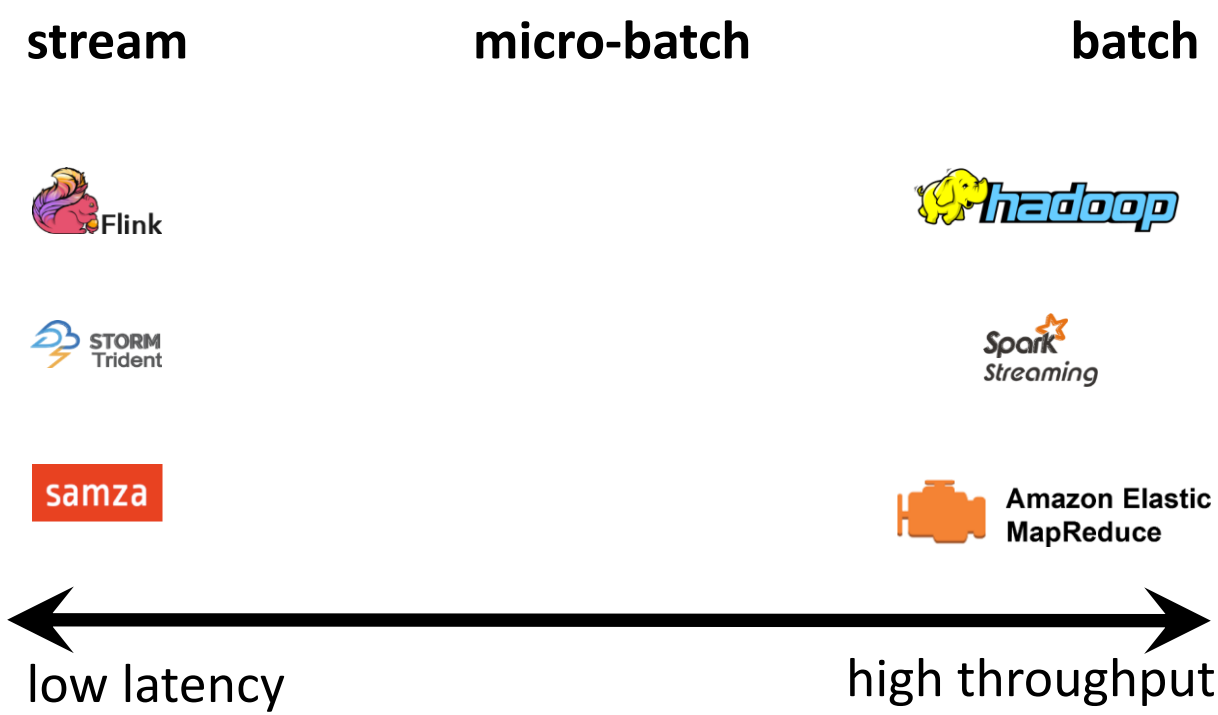
Divide and Conquer



Data Processing Frameworks



Processing Models



MapReduce / Hadoop

Google's MapReduce paradigm for distributed computing has spread widely through the open-source implementation Hadoop, offering:

- Simple parallel programming model
- Straightforward scaling to hundreds/thousands of machines.
- Fault tolerance through redundancy

Components of Hadoop

Core Hadoop has two main systems:

- **Hadoop / MapReduce**: distributed big data processing infrastructure (abstract / paradigm, fault-tolerant, schedule, execution)
- **HDFS (Hadoop Distributed File System)**: fault-tolerant, high-bandwidth, high availability distributed storage

Ideas Behind MapReduce

- **Scale “out”** (not “up”): recognize limits of large shared-memory machines and distribute work accordingly
- **Move processing to the data**: clusters have limited bandwidth, so move processes (code) rather than data
- **Avoid random access** (i.e. process data sequentially): seeks are expensive, disk throughput is reasonable
- **Seamless scalability**: from the mythical person-month to the tradable machine-hour

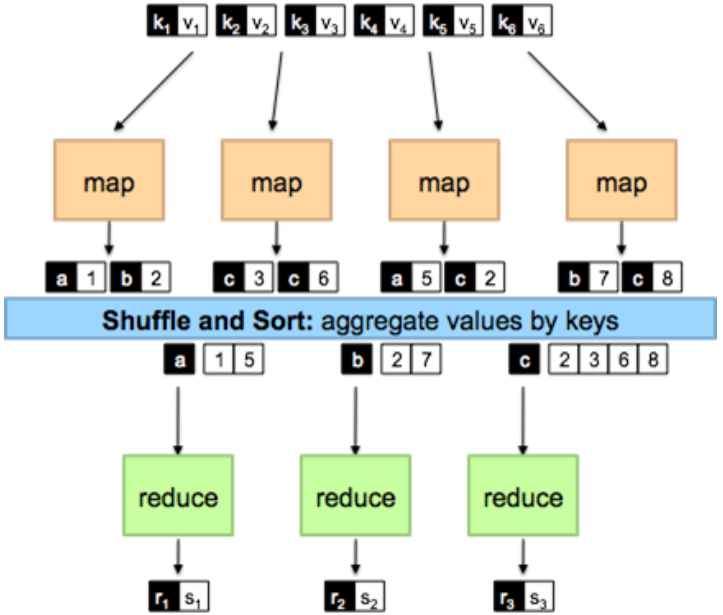
Map and Reduce

Programmers specify two functions:

map (k, v) → [(k', v')]

reduce (k', [v']) → [(k', v')]

All values with the same key are sent to the same reducer



MapReduce Word Count

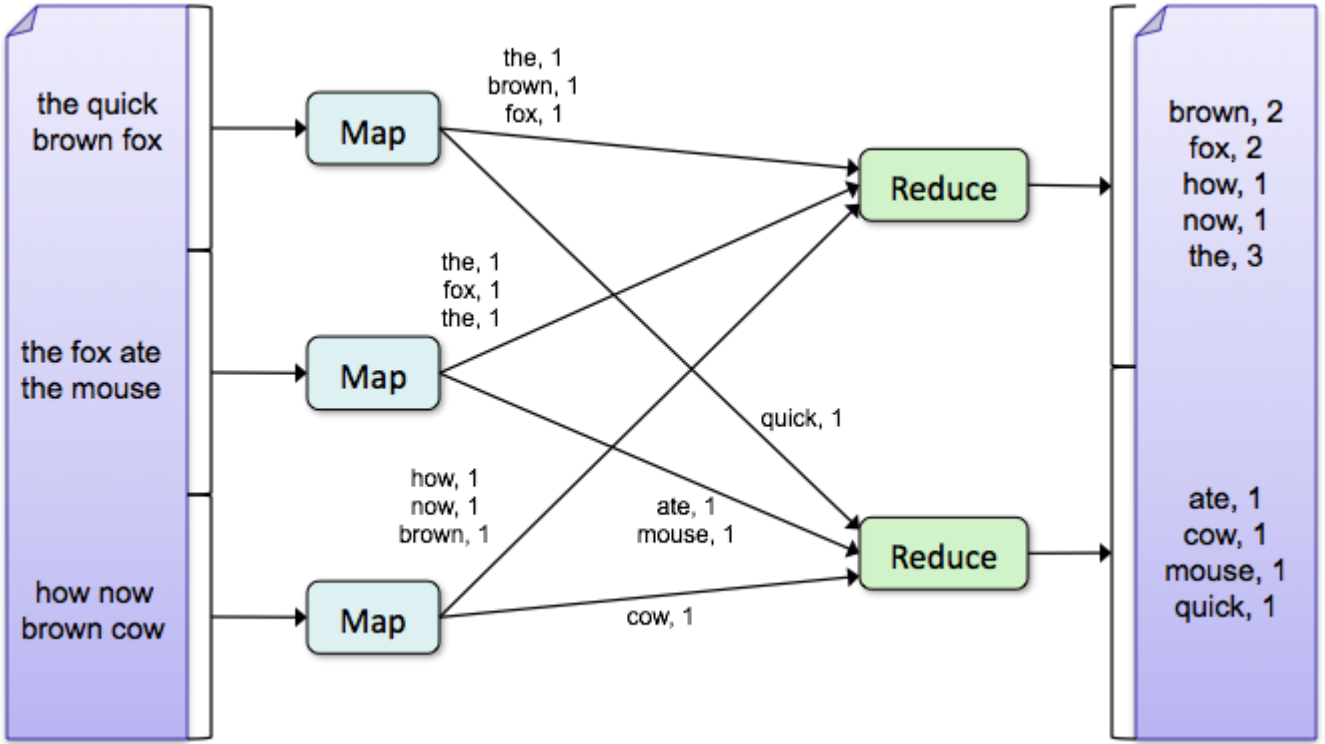
Map(String docid, String text):

```
    for each word w in text:
        Emit(w, 1);
```

Reduce(String term, Iterator<Int> values):

```
    int sum = 0;
    for each v in values:
        sum += v;
    Emit(term, sum);
```

Word Count Execution



Other Programming Primitives

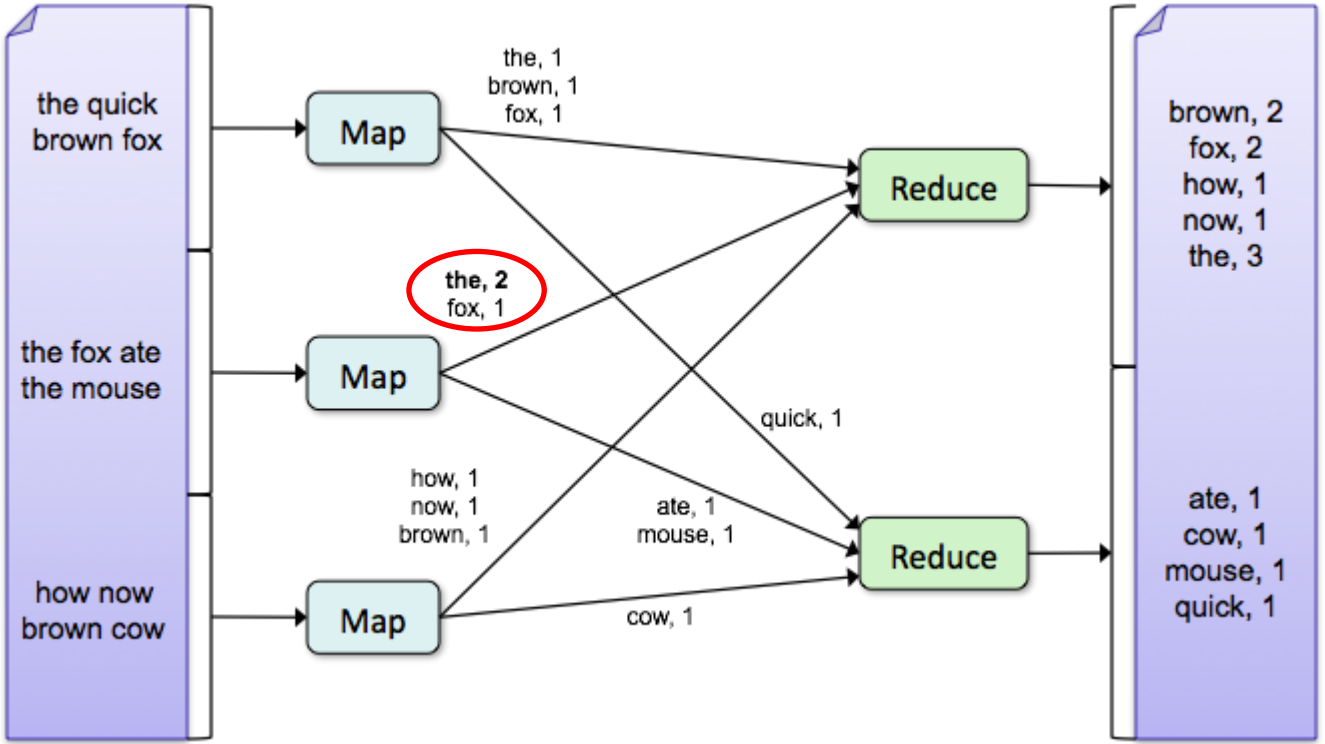
partition (k', number of partitions) → partition for k'

- Often a simple hash of the key, e.g., hash(k') mod n
- Divides up key space for parallel reduce operations

combine (k', [v']) → [(k', v'')]

- Mini-reducers that run in memory after the map phase
- Used as an optimization to reduce network traffic

Word Count with Combiner



Cloud Computing Services

Platforms like Amazon make it easy to rent large numbers of machines for short-term jobs.

There are charges on bandwidth, processors, memory, long-term storage: making it non-trivial to price exactly.

Spot pricing and reserved instances lower costs for special usage patterns.

Feel Free to Experiment

Micro instances are only 1GB, single processor virtual machines.

Reasonable machines rent for 10 to 30 cents/hr.

Free Tier*

As part of AWS's Free Usage Tier, new AWS customers can get started with Amazon EC2 for free. Upon sign-up, new AWS customers receive the following EC2 services each month for one year:

- 750 hours of EC2 running Linux, RHEL, or SLES t2.micro instance usage
- 750 hours of EC2 running Microsoft Windows Server t2.micro instance usage
- 750 hours of Elastic Load Balancing plus 15 GB data processing
- 30 GB of Amazon Elastic Block Storage in any combination of General Purpose (SSD) or Magnetic, plus 2 million I/Os (with Magnetic) and 1 GB of snapshot storage
- 15 GB of bandwidth out aggregated across all AWS services
- 1 GB of Regional Data Transfer

Ethics and AI

eth·ics
/ˈeTHiks/

noun

1. moral principles that govern a person's behavior or the conducting of an activity.
   "medical ethics also enter into the question"
   synonyms: moral code, morals, morality, values, rights and wrongs, principles, ideals, standards (of behavior), value system, virtues, dictates of conscience
   "your so-called newspaper is clearly not burdened by a sense of ethics"
2. the branch of knowledge that deals with moral principles.

- People build and apply technology, so the ethical demands are upon those who build and use AI systems.
- Moral principles are often in tension with each other, and different people can reasonably hold to different beliefs and standards.

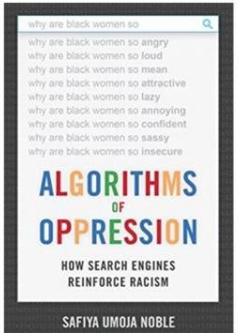
Ethical Concerns about ML Models

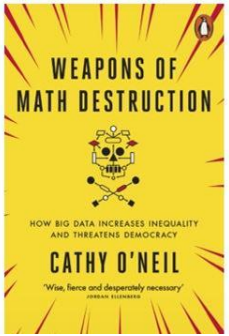
Economic dislocation: will these machines take jobs from people?

Privacy concerns: is too much private data being given to machines to build these models?

Bias concerns: are these models learning the wrong thing from the training data?

Responsibility issues: are people adequately in the decision loop to step in when models go astray?





Privacy Concerns

The power of Big Data is such that large companies gather private information on a scale many people find threatening: Google, Facebook, Amazon / Alexa, OpenAI, ...

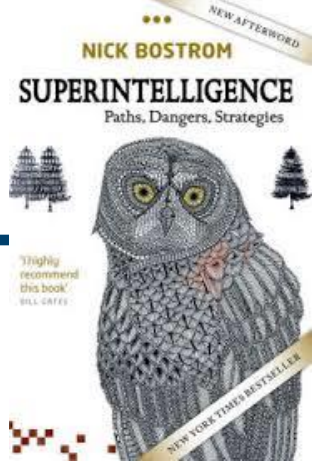
- Face identification means ubiquitous video surveillance.
- Conversations with may leak sensitive information.
- More subtle concerns about what inferences are being made about you: e.g. models that predict whether you are gay or pregnant from observable data.

Bias Concerns

Models trained on biased data learn these biases
(e.g. Amazon's resume screener, Google's image search)
It is hard to build accurate classifiers for small/minority classes.
But:

- people are also biased → it should be easier to evaluate algorithms than people. Biased AI is usually accidental.
- fairness is often hard to define: e.g. do you want the system that minimizes errors globally or equalizes all groups?

Responsibility Issues



What should we automate and how?!

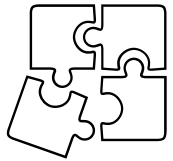
- The real problem is people trusting AI models too much.
- Should algorithms be permitted to fire weapons in combat?
- When is a self-driving car safe enough to drive?
- Are mistakes correctable? → Who do you complain to when a program eliminates your resume before a human ever sees it?
- Are models correctable? → Are processes in place which continually evaluate models and improve them with time?
- Ultimately these are human decisions, and we must pass judgment on how to use AI or any other technology.

Economic Dislocation

Blue collar jobs are threatened: drivers, security guards,…
White collar work are threatened: lawyers, translation/transcription, education, pathology/medicine…
But: technology has always **destroyed jobs**, yet generated new and **better jobs**, in ways impossible to predict.
The world will always change rapidly and you'll have to adapt.

You are part of this development!

Wrapup: Data Science in the Wild



- Achieving scale can be a significant challenge to data science in practice
  → data engineering („How can this be done?“)
- Data science is embedded into actual problems in our society
  → ethics, economics, privacy, ...
  („How should this (not) be done?“)