Introductory session

Big Data, Machine Learning &
Data Science

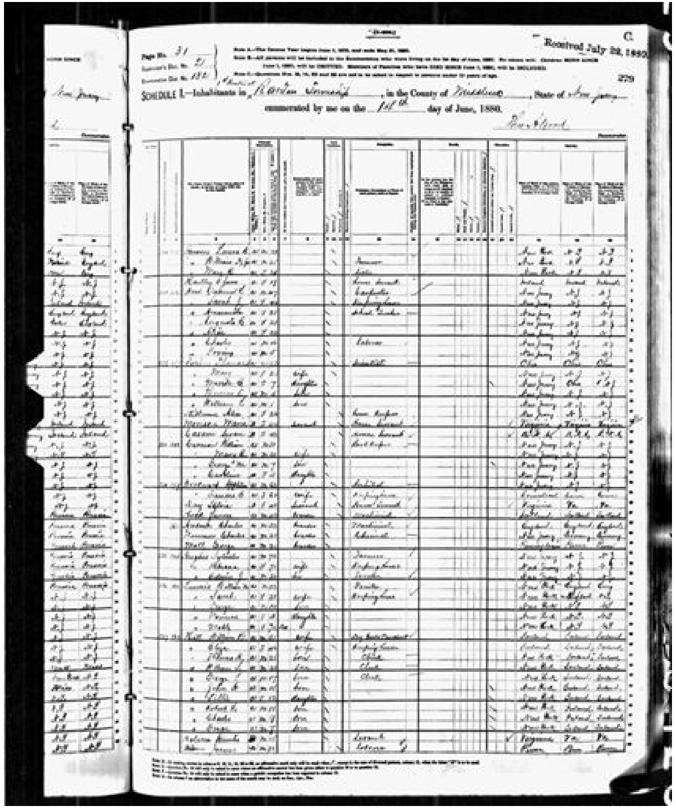
A history of (Big) Data

An IDC estimate […] is forecasting a tenfold growth by 2020 to 44 zettabytes.

A zettabyte is 10^{21} bytes, or equivalently one billion terabytes.

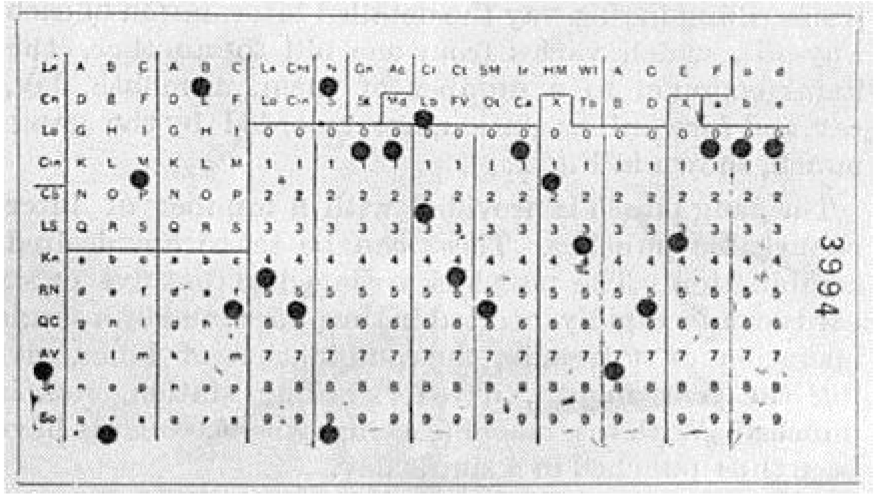
That's more than one disk drive for every person in the world.

1880 : 1890 - the US Census



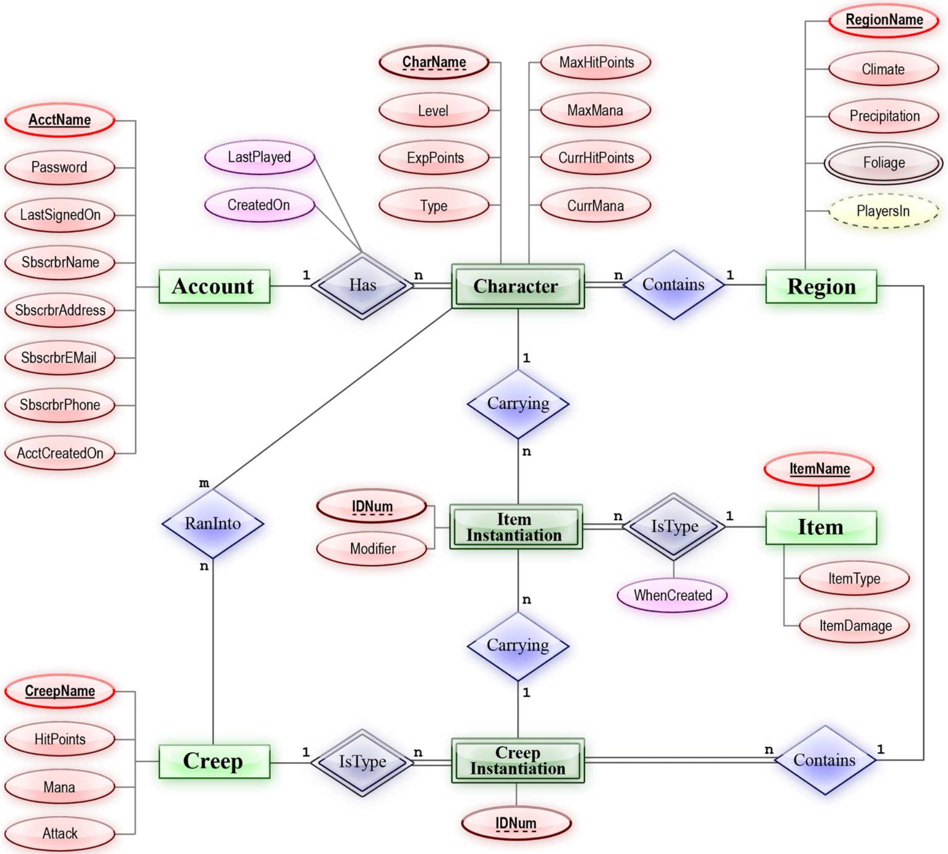
- Enumeration sheets
- Hard to access on machines
- 50 million people
- 8 years to analyze
- Obsolete results 😞

1880 : 1890 - the US Census



- Punched cards
- Readable by tabulating system
- 63 million people
- 6 years to analyze instead of estimated 11

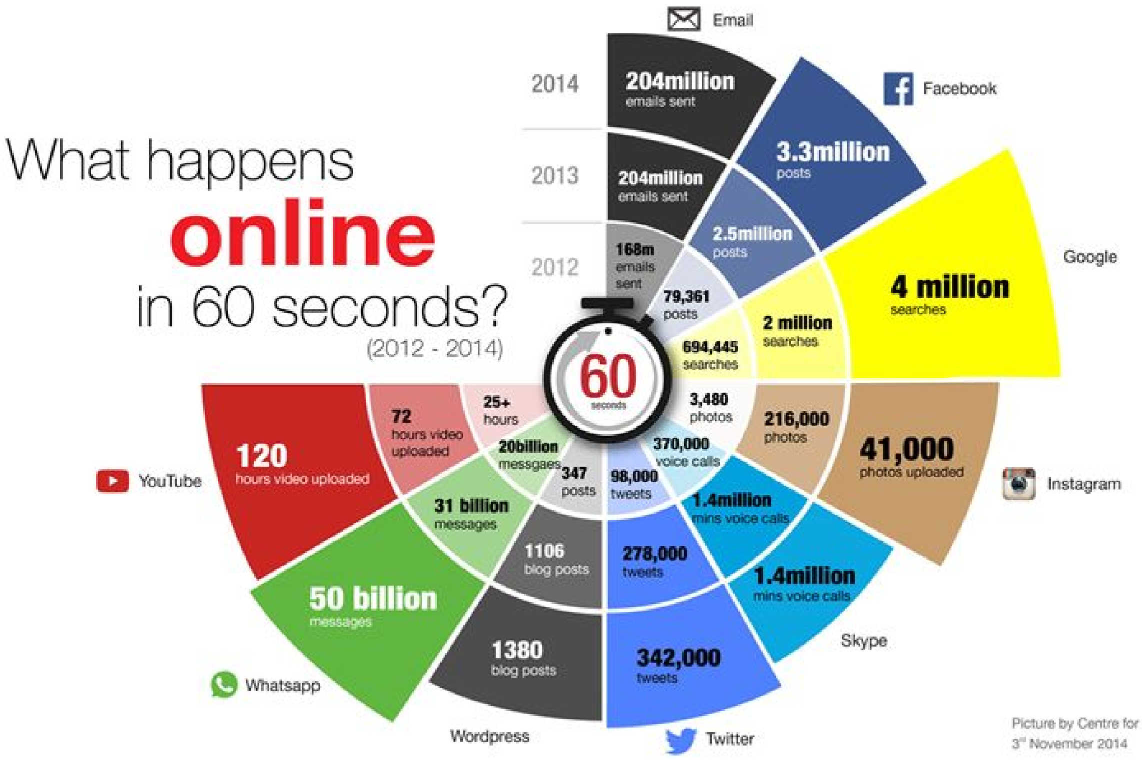
1970 : 1990 - Creating manageable data structures



- 1970s : invention of the relational data model and relational database management system (RDBMS)
- 1976 : Entity-relationship modeling
- Around 1980s : Object database management system
- 1990s : Commercialization of the first data warehouses

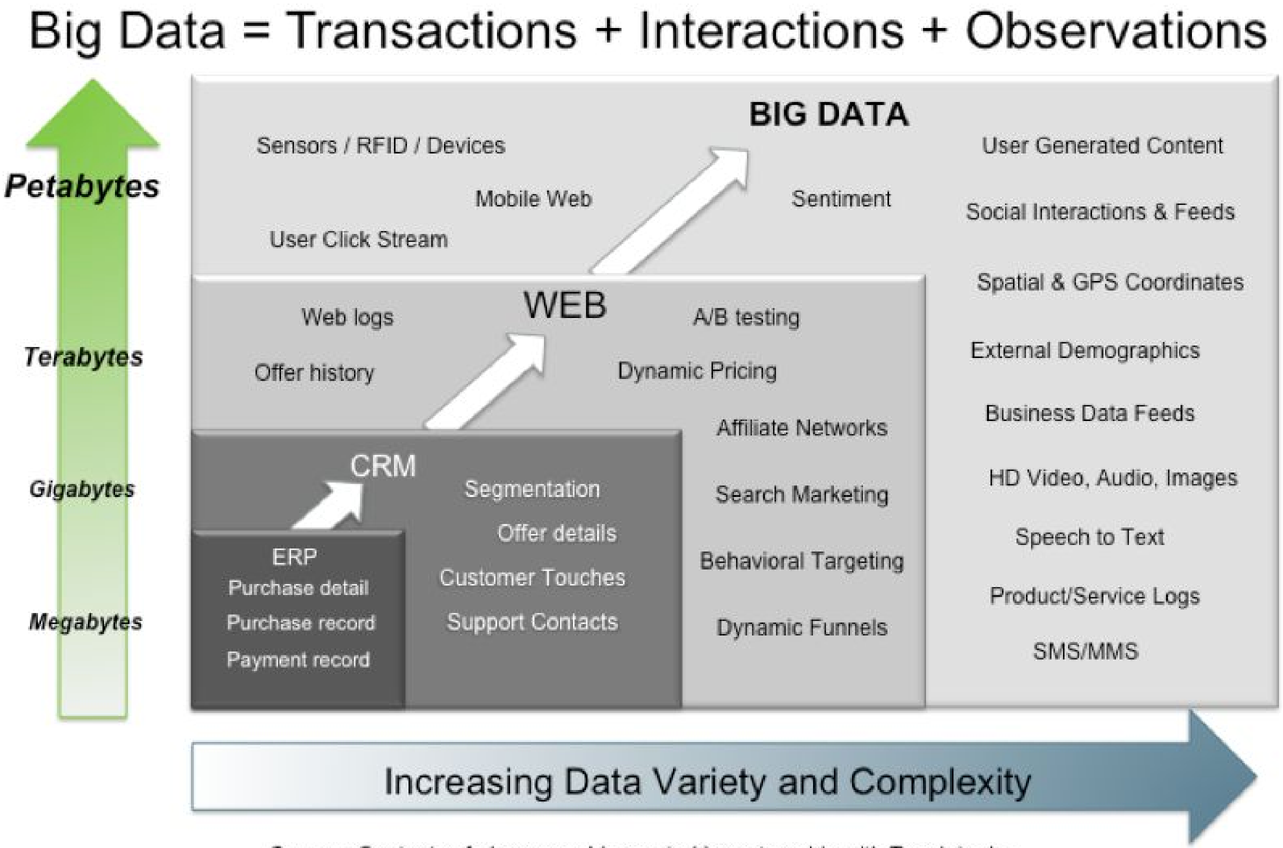
2000s - Web and content management

Rise of the web → manage unstructured data : web content, images, audio, video…



Picture by Centre for Learning and Teaching
3rd November 2014

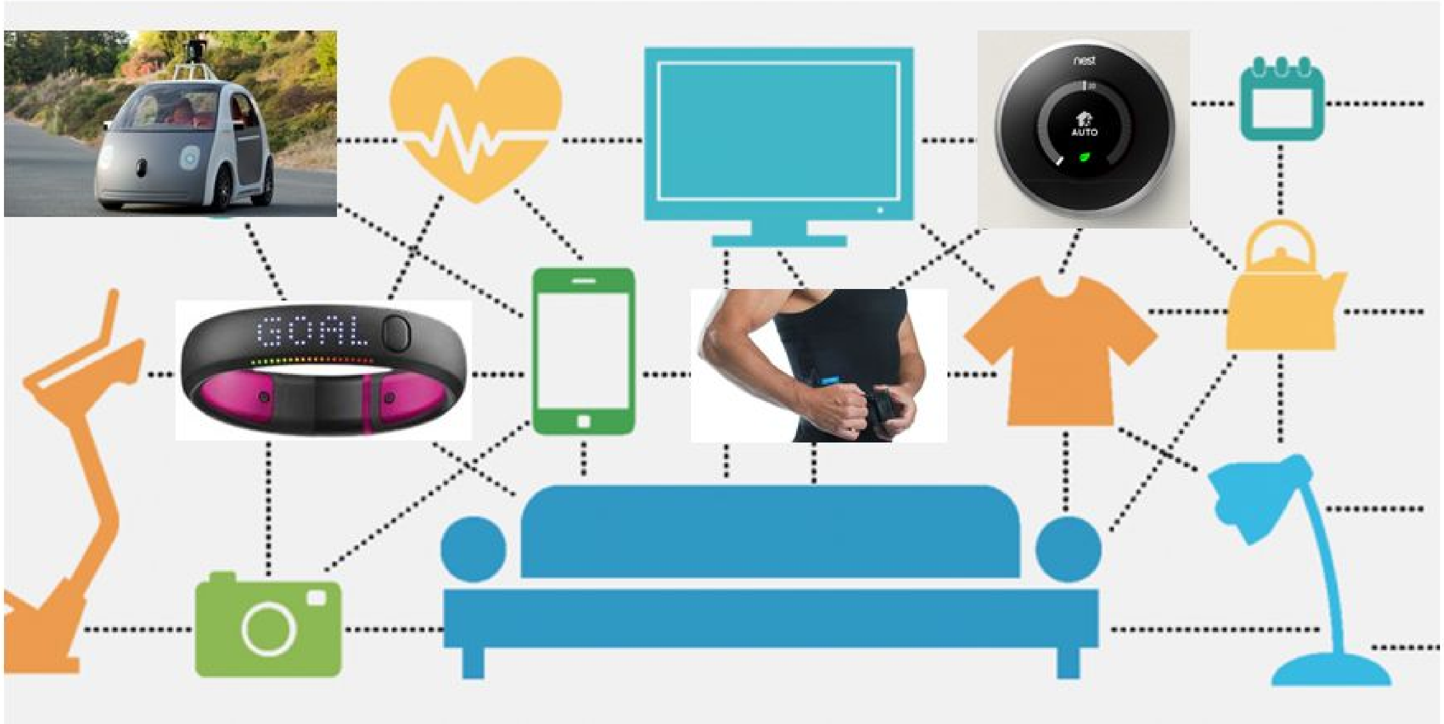
Today



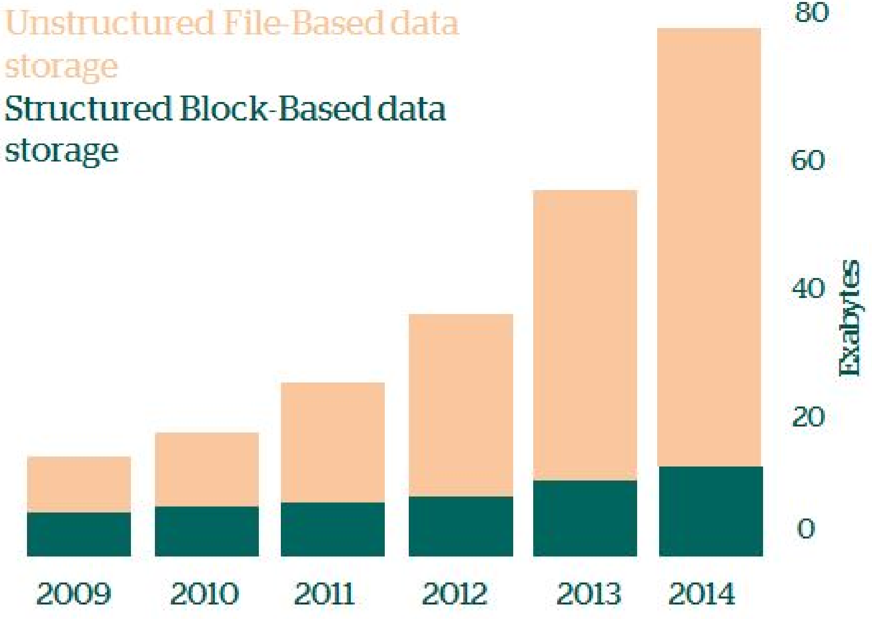
Source: *Contents of above graphic created in partnership with Teradata, Inc.*

Tomorrow ?

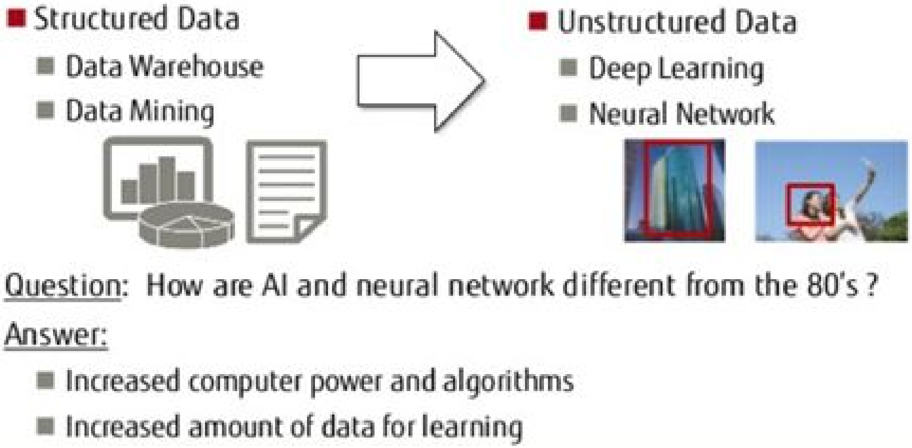
The emerging Internet of Things makes every thing a data or content, adding billions of sources of data to the overall picture.



Data deluge 101 - to collect is human



Rapid growth of data in the digital universe



Changes in Data intensive Computing

Distributed computing for Big Data

Grace Hopper - In pioneer days they used oxen for heavy pulling,

and when one ox couldn’t budge a log, they didn’t try to grow a larger ox.

Too much data

Today we have storage and computation needs that don’t fit on a single machine

April 10, 2014: Facebook’s warehouse stores upwards of 300 PB of data, with an incoming daily rate of about 600 TB.

In the last year, growth in amount of data stored has tripled.

September 2, 2015: Linkedin hit a record by processing a trillion messages per day, with peaks of 4,5 million messages per second.

That’s 1,34 PB information per week and a 1.200x growth in 4 years.

How do we solve that problem ?

To distributed computing

Scaling up		Scaling out	
			
Less power consumption, cooling costs Less challenging to implement Less licencing costs (Sometimes) less network hardware	**PRICE** Hardware failure causes bigger outages Vendor lock-in Limited upgradeability	Much cheaper Easier fault-tolerance “Easier” upgrade by adding new machines	Bigger energy footprint Higher utility cost (electricity, cooling) More networking equipment

To distributed computing

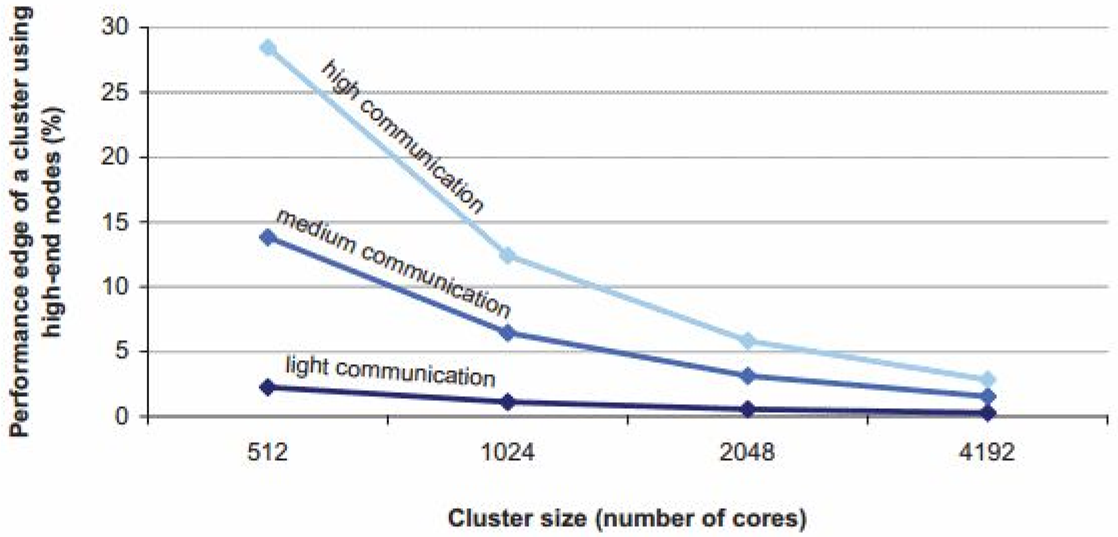
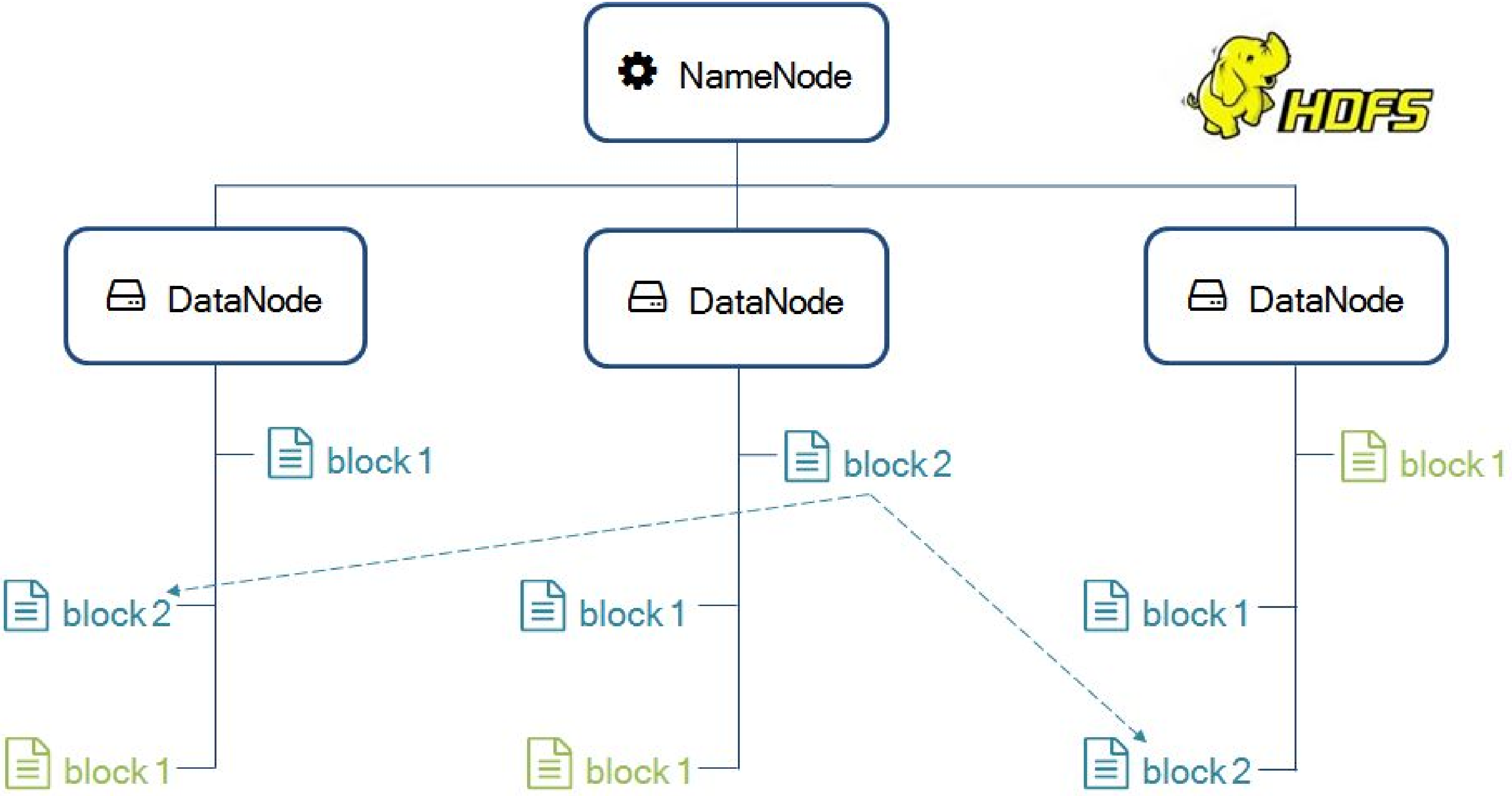


FIGURE 3.2: Performance advantage of a cluster built with high-end server nodes (128-core SMP) over a cluster with the same number of processor cores built with low-end server nodes (four-core SMP), for clusters of varying size.

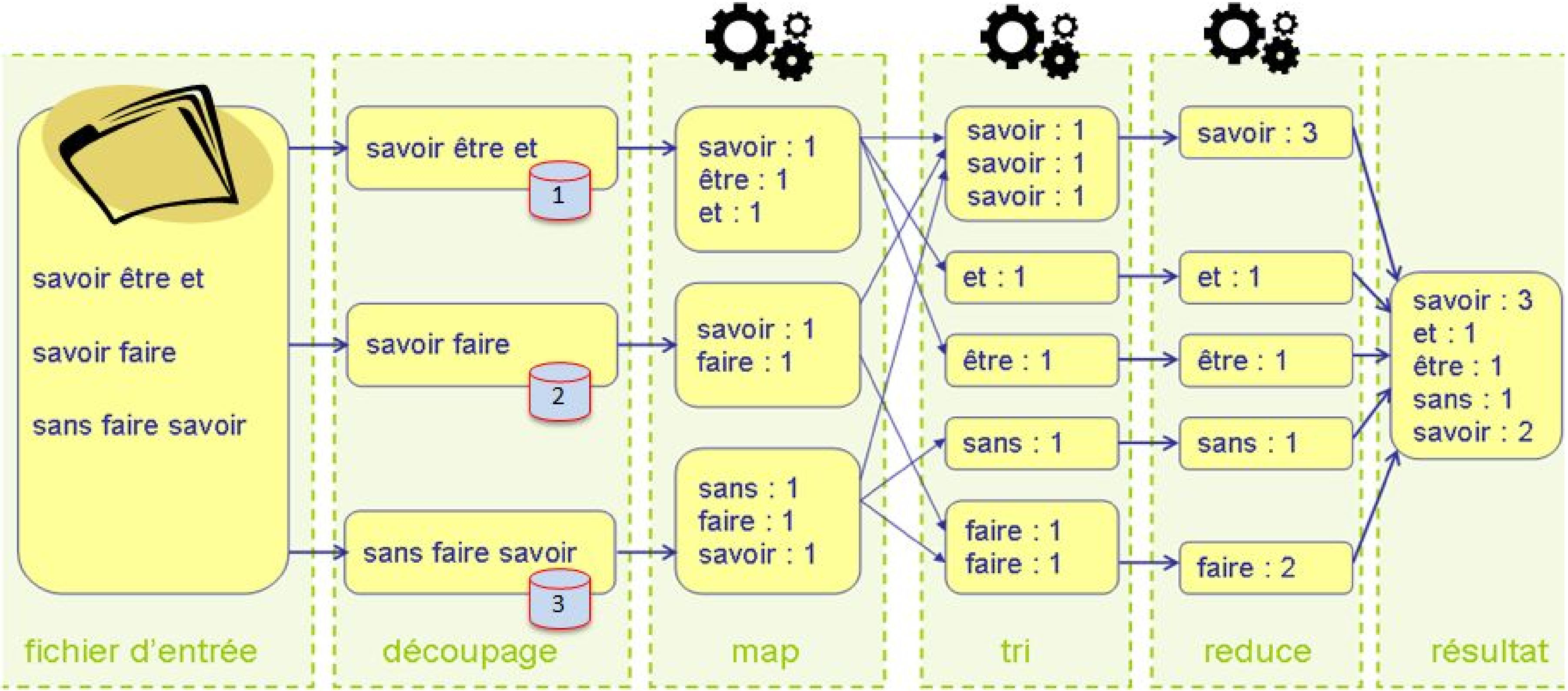
The performance gap between high-end and commodity hardware decreases with cluster size (assuming a uniform memory access pattern across all nodes).

Distributed data storage



Distributed data processing

Moving computation is cheaper than moving data



Big Data 101

Distributed systems **IS HARD** :

- Knowledge is local, any information on global state is potentially out of date
- Nodes can fail / recover from failure independently
- Messages can be delayed/lost
- clocks are not synchronised accross nodes
- ...and a lot of other scary stuff

-.-

Big Data 101

...but distributed systems is a cheap and efficient solution to cope with the extremely higher demand of users in both processing power and data storage.

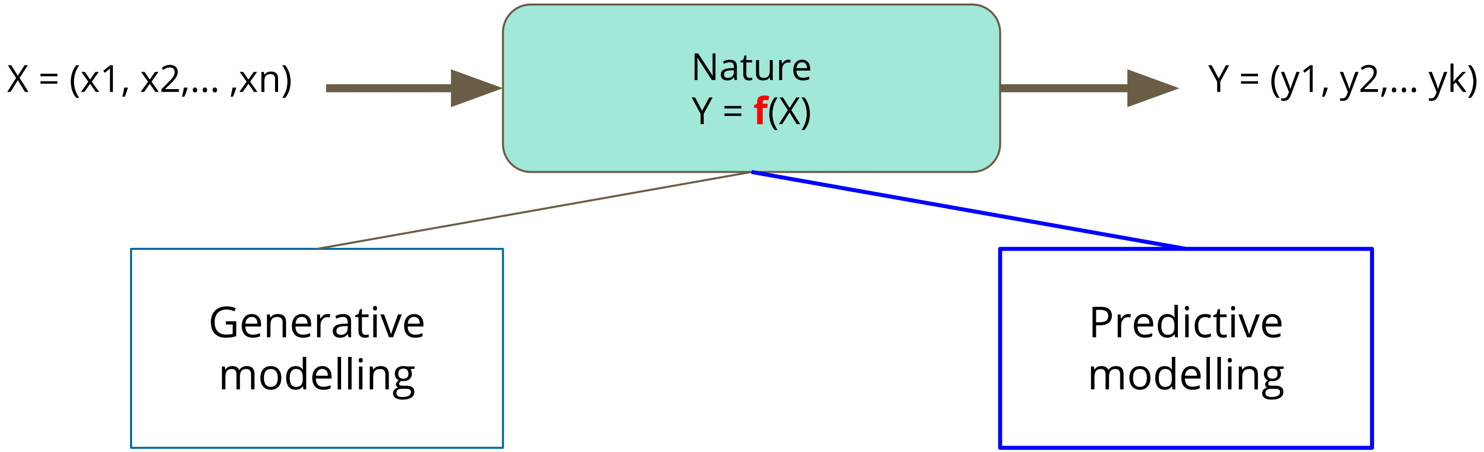
It is about raising the level of abstraction, to create building blocks for programmers who have lots of data to store and analyse and not experts in distributed systems

Machine Learning

Hadley Wickham - Uncovering Truth is extremely difficult,

and even if possible, maybe so complicated as to not be practically useful.

Statistical learning



- Look for the true model
- Test hypothesis, confidence intervals, relationship measurement
- Traditional Academics statistics

- Silent about underlying mechanism
- Focus on predictive accuracy
- “Industrial” statistics : methods are off-the-shelf while still incorporating human knowledge

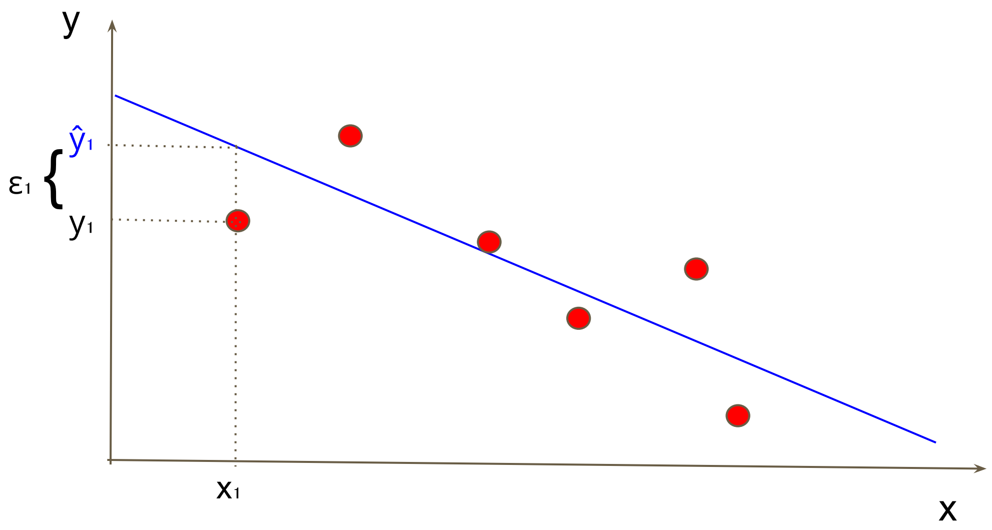
Example - Linear regression

Given $(x_1, y_1), (x_2, y_2) \ldots (x_n, y_n)$

Goal : find the best linear mapping f_w

$$\boxed{y \approx \hat{y} = f_w(x) = w_0 + w_1 x}$$

By fitting on the weights (w_0, w_1)



We need to evaluate predictive accuracy between our predictions and the truth

Squared vertical error

$\varepsilon_k = (y_k - \hat{y}_k)^2$

+

sum error on all points

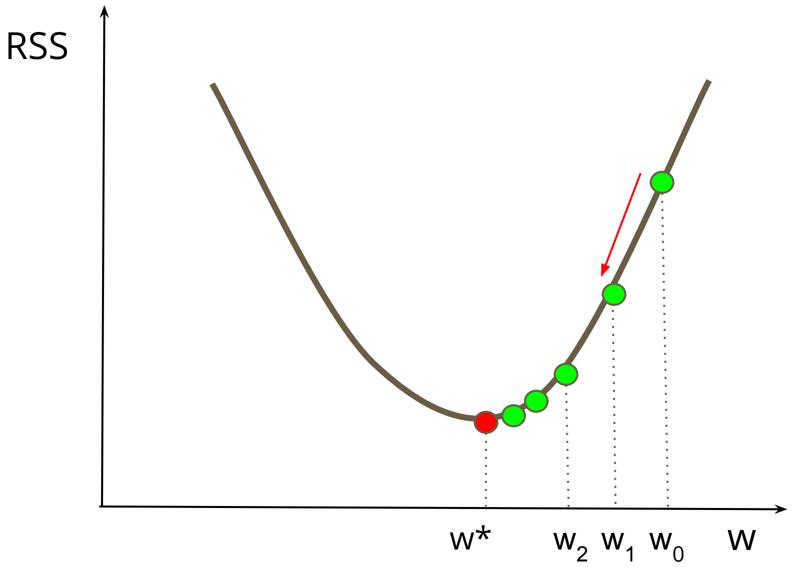
compute w to minimize

$$\min_w \sum_{i=0}^{n} (y_i - f_w(x_i))^2$$

Example - Linear regression

Find w* that minimizes $$RSS(w) = \sum_{i=0}^{n} (y_i - f_w(x_i))^2 = \sum_{i=0}^{n} (y_i - w_0 - w_1 x)^2$$

Iterative solution - Gradient descent



Update rule :

$$w_{i+1} = w_i - \alpha_i \frac{\partial RSS}{\partial w}(w_i)$$

Intermediate conclusion

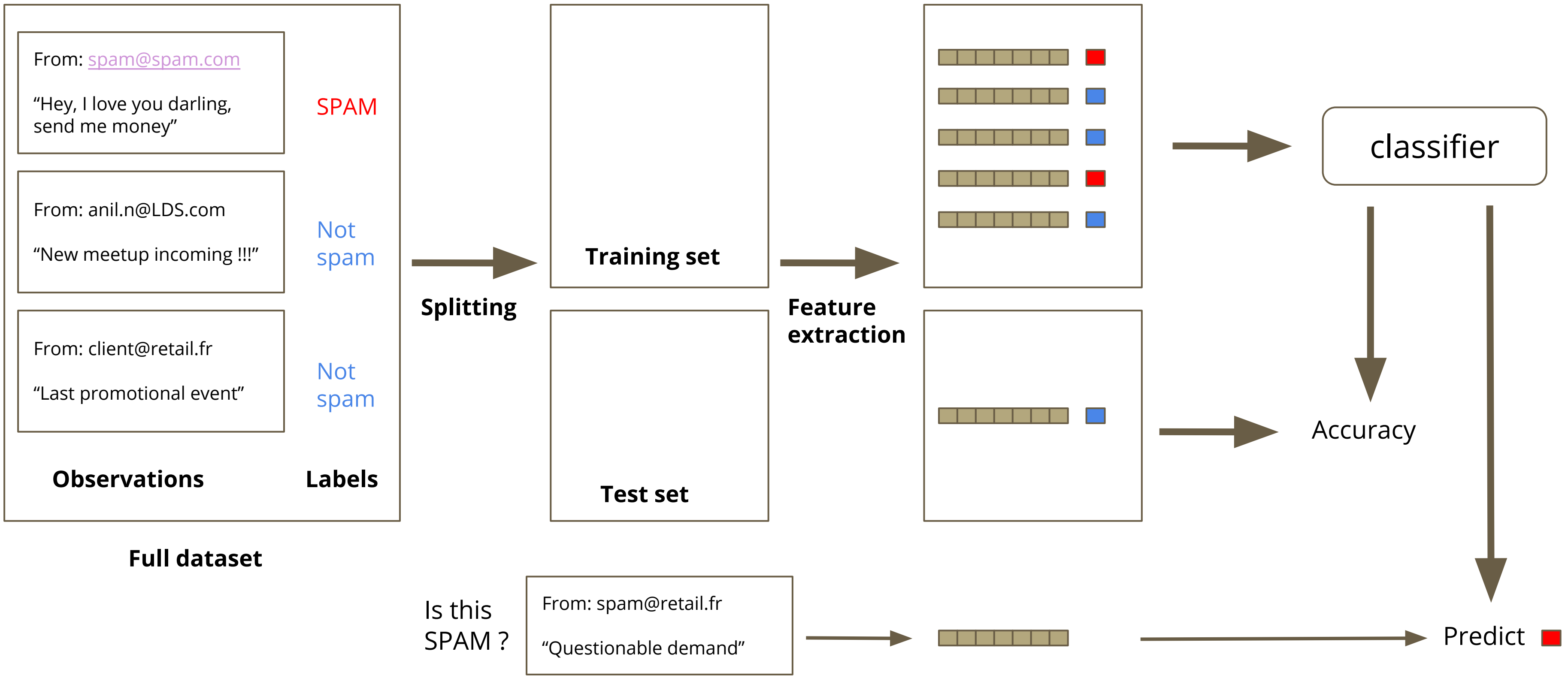
A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E.

Defining the task T (supervised, unsupervised, transcription, anomaly detection...)	Choosing the performance P (accuracy, error rate...) to measure on a **test** dataset	Collecting the experience E as a **training** set, to optimize T in order to maximize P on it.
$\hat{y} = \boldsymbol{w}^\top \boldsymbol{x}$	$\mathrm{MSE}_{\mathrm{test}} = \frac{1}{m} \sum_i (\hat{\boldsymbol{y}}^{(\mathrm{test})} - \boldsymbol{y}^{(\mathrm{test})})_i^2.$	$(\boldsymbol{X}^{(\mathrm{train})}, \boldsymbol{y}^{(\mathrm{train})})$

Improve the weights w in a way that reduces $\mathrm{MSE}_{\mathrm{test}}$ when the algorithm gains experience on a training dataset.

Advanced spoiler : statistics field gives us the foundations to formally deal with the learning problem, and its tendency to generalize or not.

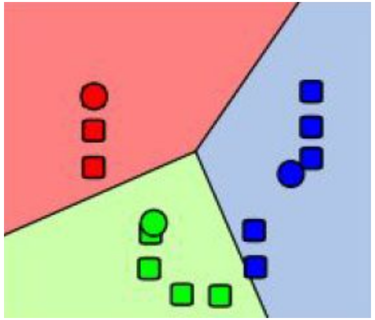
Terminology & basic pipeline



Supervised learning

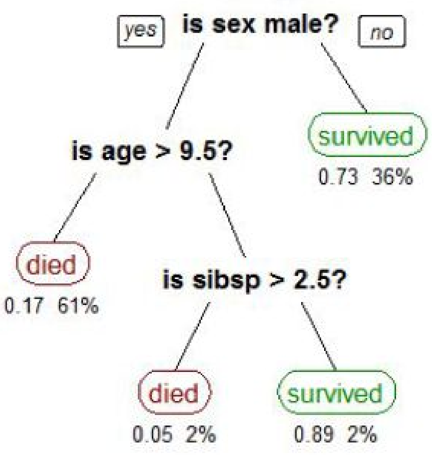
Learning from labeled observations that tell you if you are “right” or “wrong”.

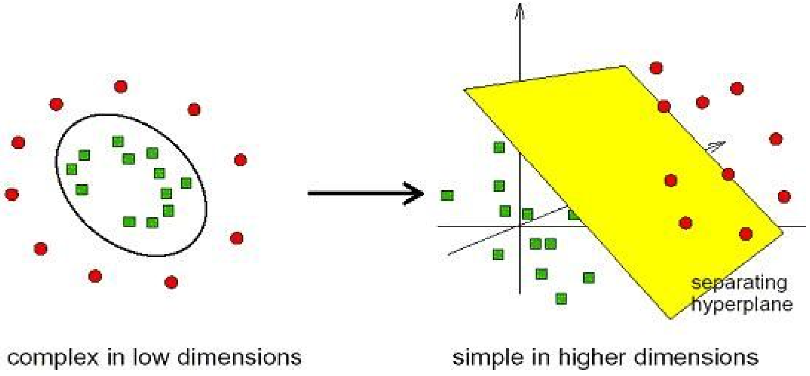
- Classification = Categorical/label prediction (e.g : is it spam or not ? what kind of flower is it ?)
- Regression = Numerical/vector prediction (e.g : how much is it going to rain ? what age is he ?)



K-Means

Decision Tree



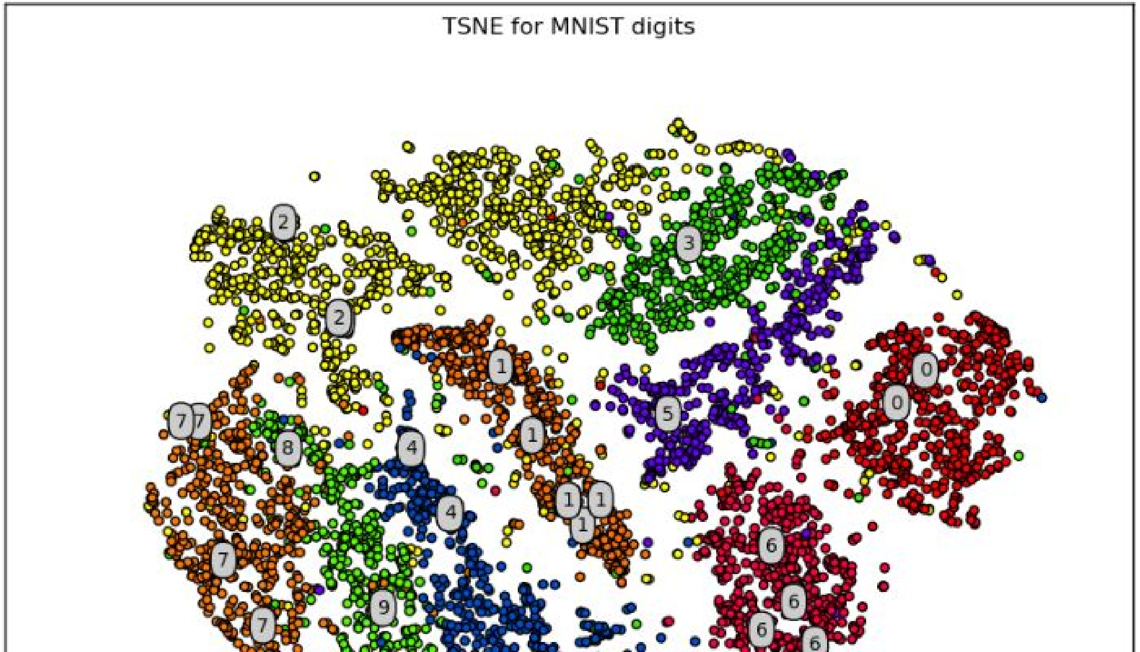


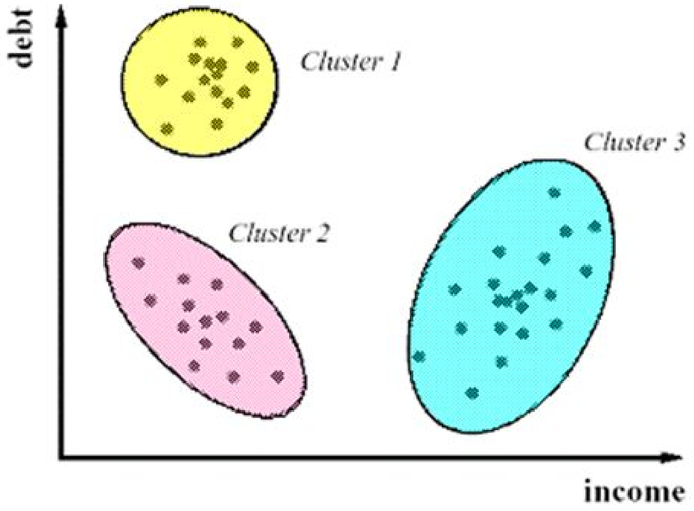
SVM

Unsupervised learning

Learning from unlabeled observations by directly inferring “latent” properties

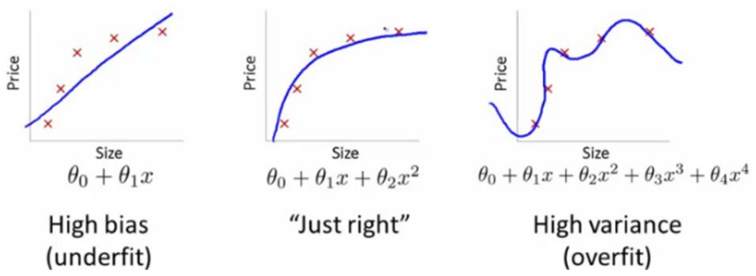
- Clustering : Categorical/label
- Dimensionality reduction : Vector, compression



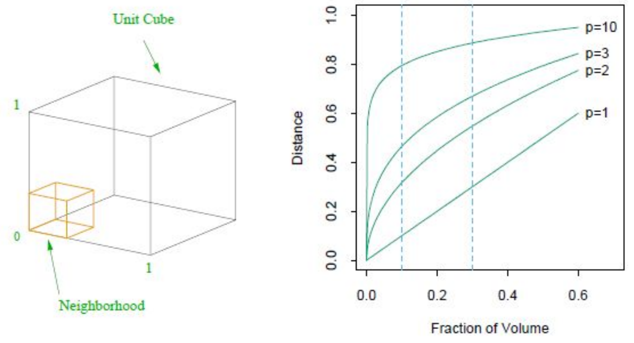


Still a lot to see ... (spoilers)

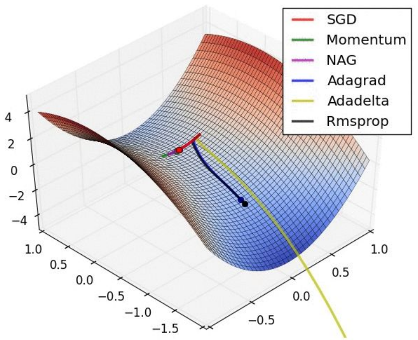
Overfitting, generalization, regularization

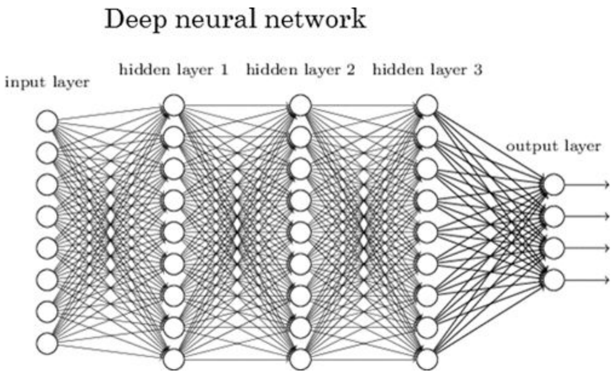


Curse of dimensionality

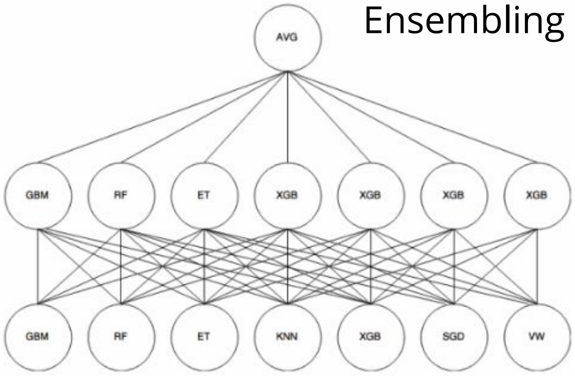


Non convex optimization





Ensembling



Data Science

David Donoho - I present a vision of data science based on the activities of people who are

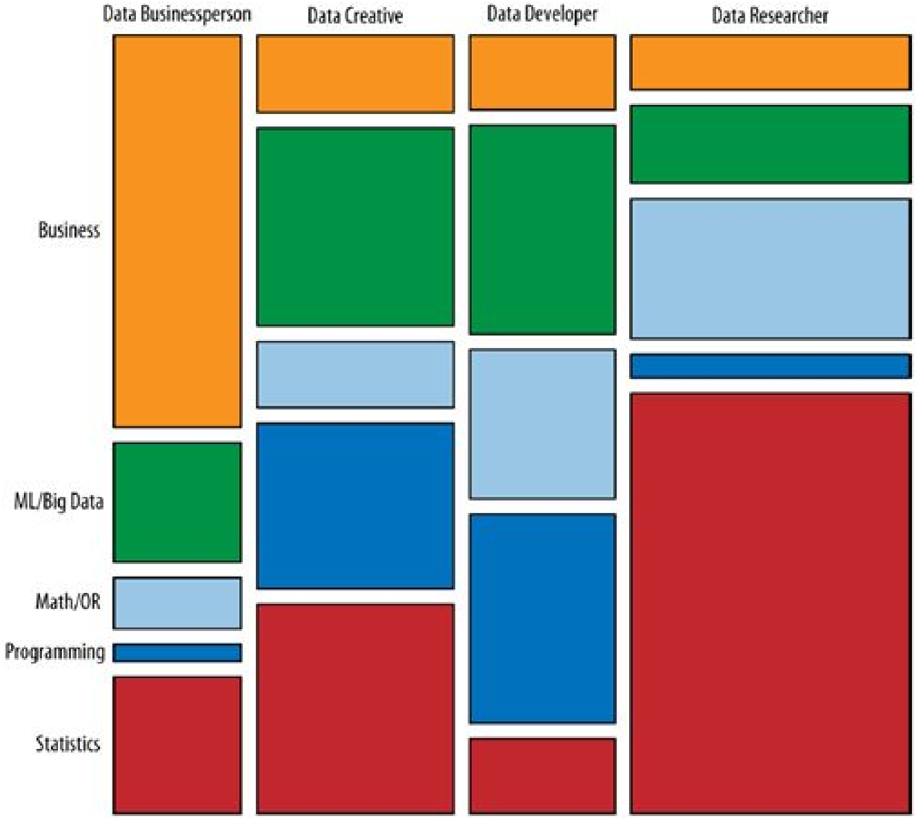
`learning from data',

with an academic field dedicated to improving that activity in an evidence-based manner.

Data science roles

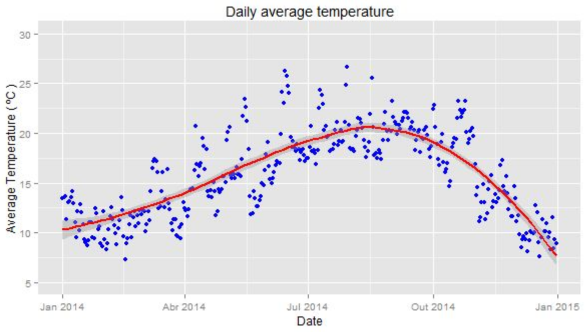
Data Developer	Developer	Engineer	
Data Researcher	Researcher	Scientist	Statistician
Data Creative	Jack of all trades	Artist	Hacker
Data Business person	Leader	Business person	Entrepreneur

http://survey.datacommunitydc.org/

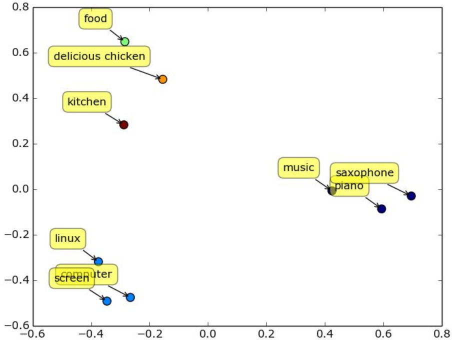


Activities of Data science

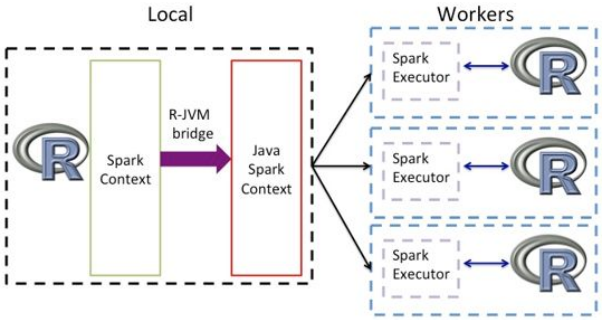
Data exploration & preparation



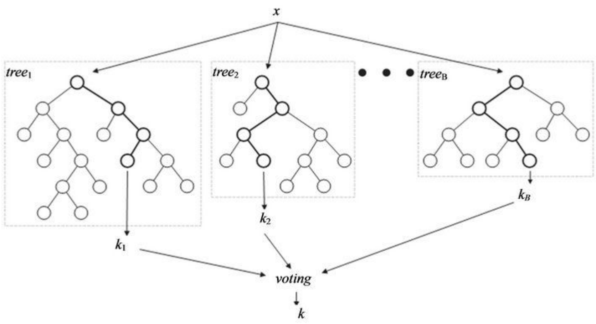
Data representation & transformation



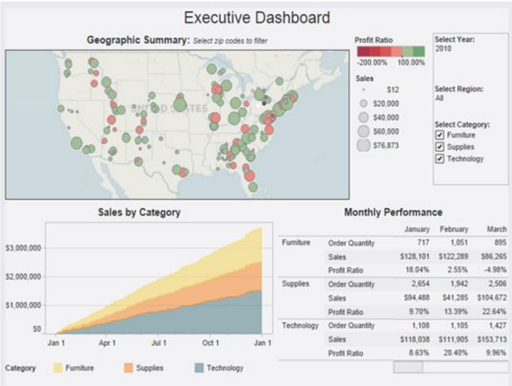
Computing with data



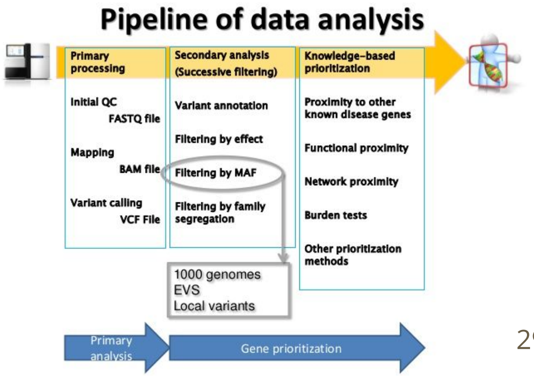
Data modelling



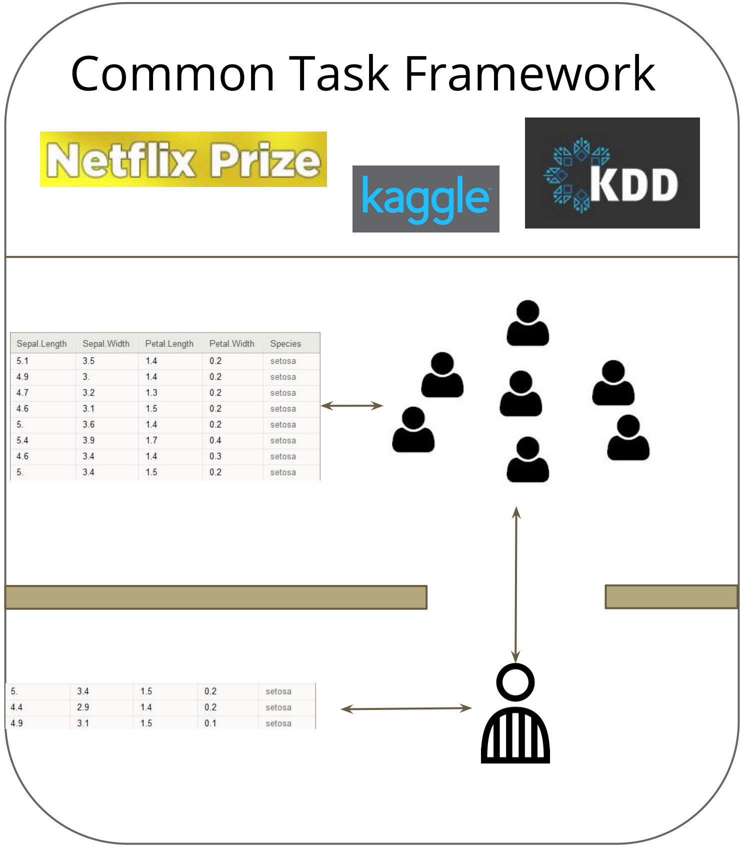
Data visualisation & presentation



Science about Data Science



The science of Data Science



Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
5.1	3.5	1.4	0.2	setosa
4.9	3.	1.4	0.2	setosa
4.7	3.2	1.3	0.2	setosa
4.6	3.1	1.5	0.2	setosa
5.	3.6	1.4	0.2	setosa
5.4	3.9	1.7	0.4	setosa
4.6	3.4	1.4	0.3	setosa
5.	3.4	1.5	0.2	setosa

5.	3.4	1.5	0.2	setosa
4.4	2.9	1.4	0.2	setosa
4.9	3.1	1.5	0.1	setosa

Total focus on optimization of empirical performance :

- Open source movement, democratization of quantitative programming environments
- Code / knowledge sharing
- Reproducible experiments, productive tweaking
- Open to anyone with IT skills
- Immediately applicable in a real world setting

Identify commonly-occuring analysis/processing workflows, improve and reuse them

The science of Data Science

Science on Data Science is a continually evolving, evidence-based approach to data analysis & predictive modelling :

- Science-Wide meta analysis, cross-study analysis, cross-workflow analysis
- Open Science and reproducible computation
- Science as data
- Empirical validation of scientific methodology

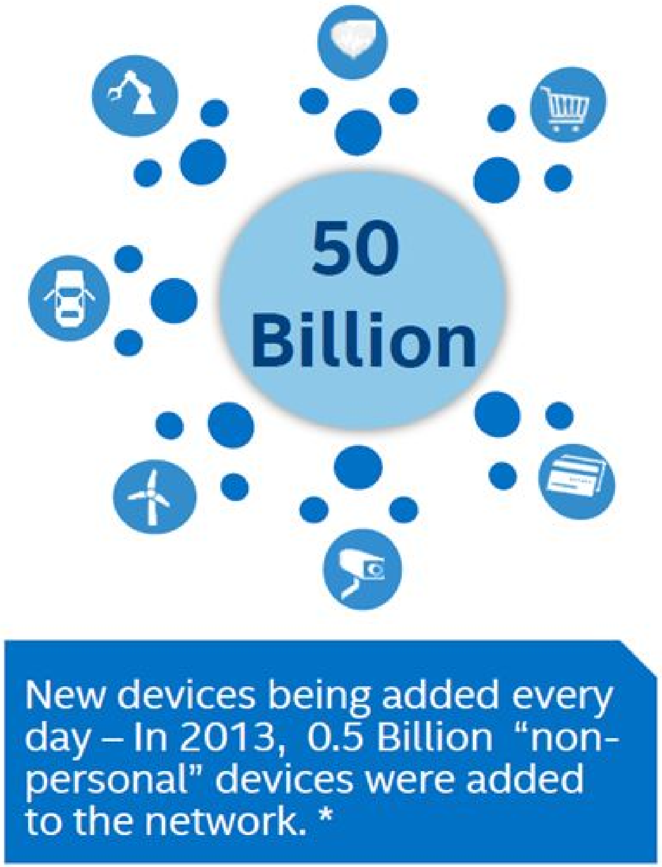
Never stop sharing your work, make it deployable & reproducible so others can tweak or build upon it.

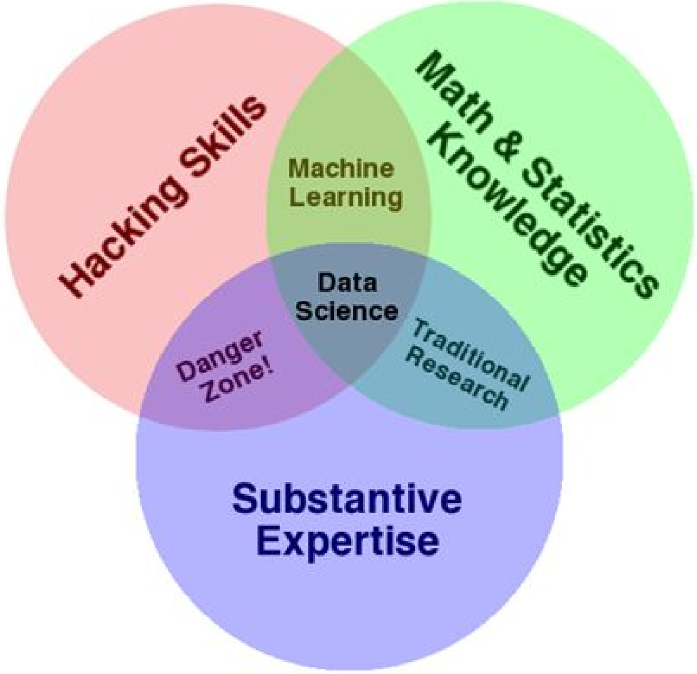
Conclusion

We all say we like data, but we don’t.

We like getting insight out of data. That’s not quite the same as liking the data itself.

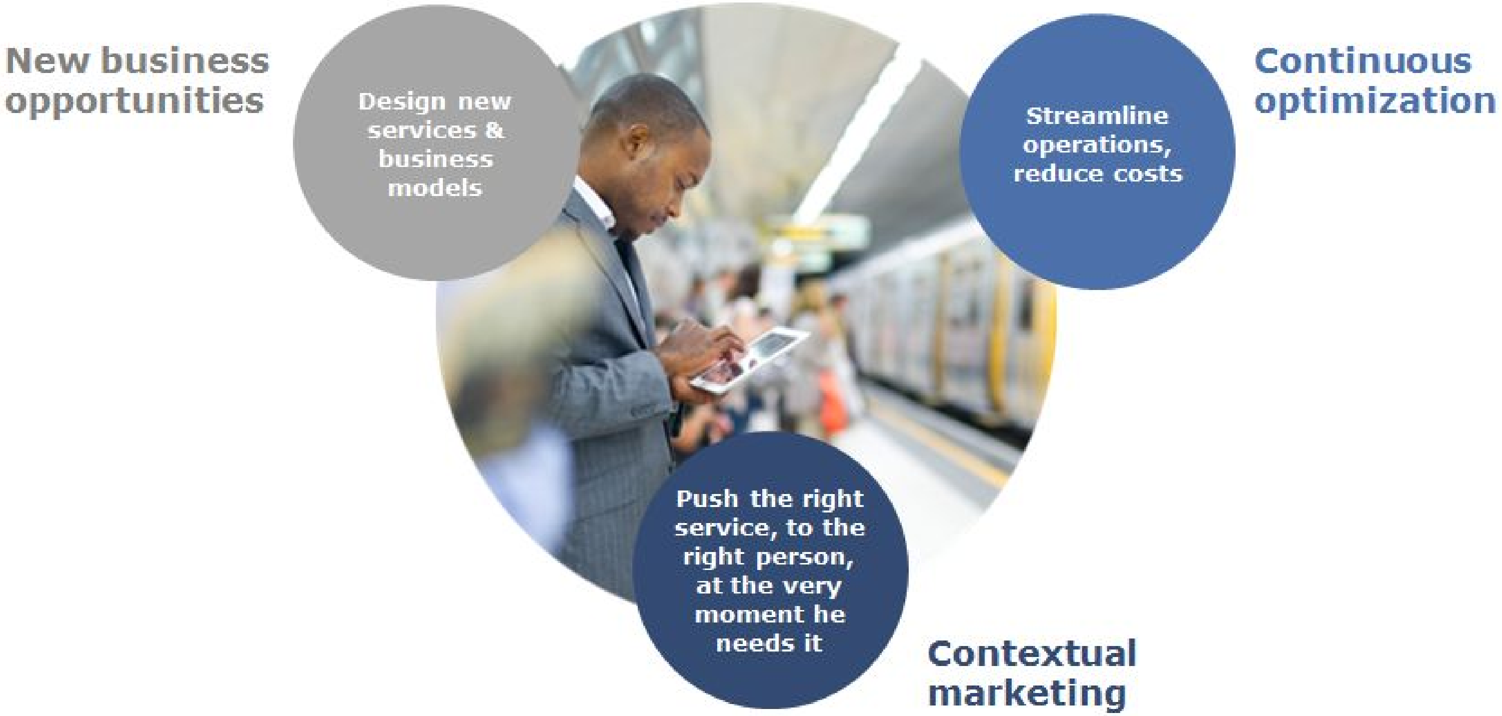
Big Data & Data science





Big Data & Data science

Data Management & Analytics is the key enabler of the brands digital transformation



Conclusion

Camille Fournier @skamille · 8h

You aren't going to hire data unicorns (science, eng, biz in one person) so get real. Word.

To leverage Big Data,
develop a cross-disciplinary team
with deep knowledge of the business with technology