Introduction to Machine Learning.
CSCI-UA 9473, Lecture 1.

Augustin Cosse

Ecole Normale Supérieure, DMA & NYU
Fondation Sciences Mathématiques de Paris.

2018

Schedule

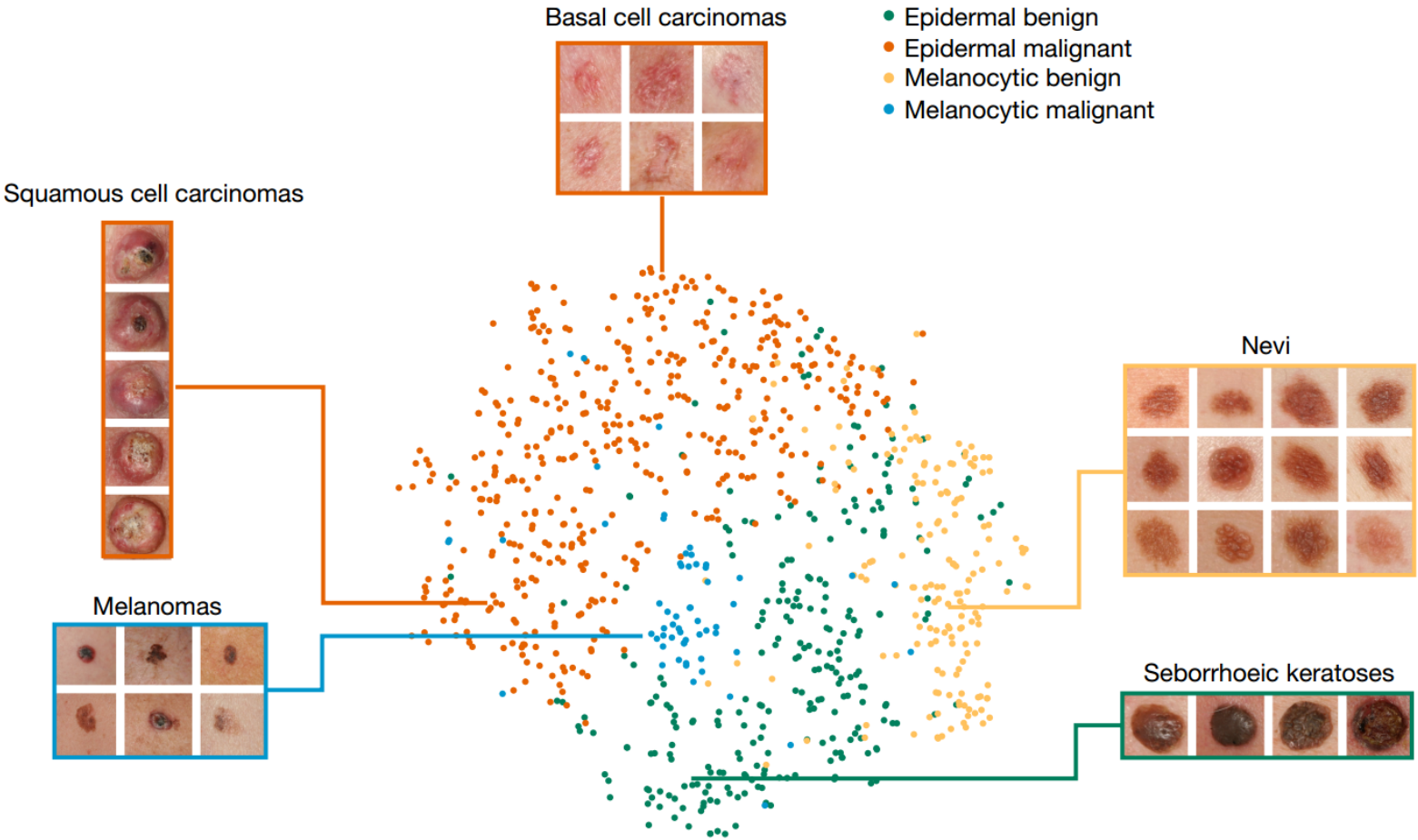
- Class and labs: Tuesday/Thursday 5.15pm to 6.45pm,
- Office hours : Tuesday/Thursday : 6.45pm to 7.15pm,
- Location: NYU Paris, 57 Boulevard Saint-Germain, Room 4.06
- Combination between programming sessions (python) and lectures
- Final Exam: Midterm: 30%, Final : 30%
- Assignements throughout the term: 30%

Today's class

- General info on the class
- Brief overview of Main techniques and challenges
- Reminders on Statistics

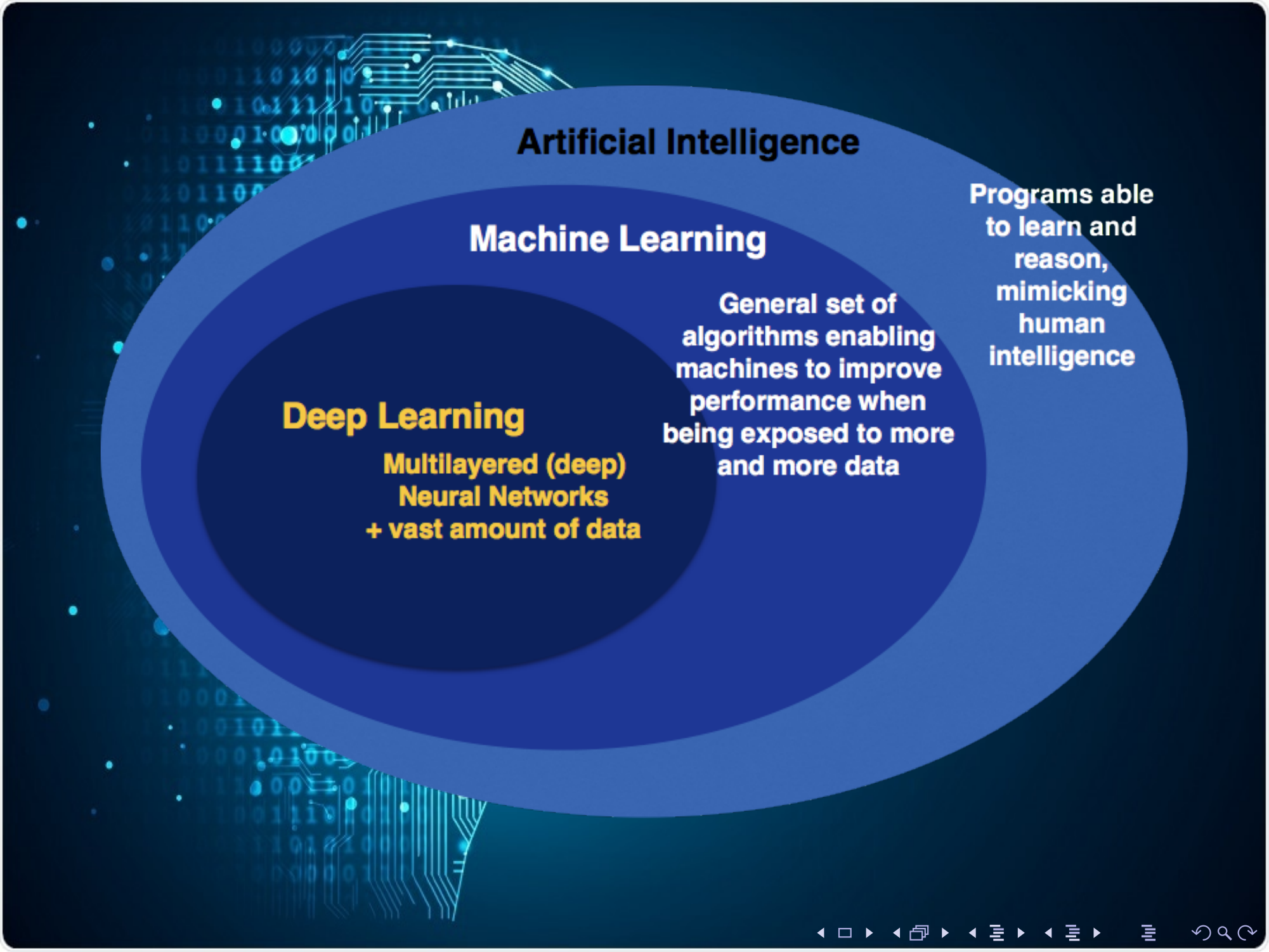
Machine learning today (I)

Machine learning today (II)

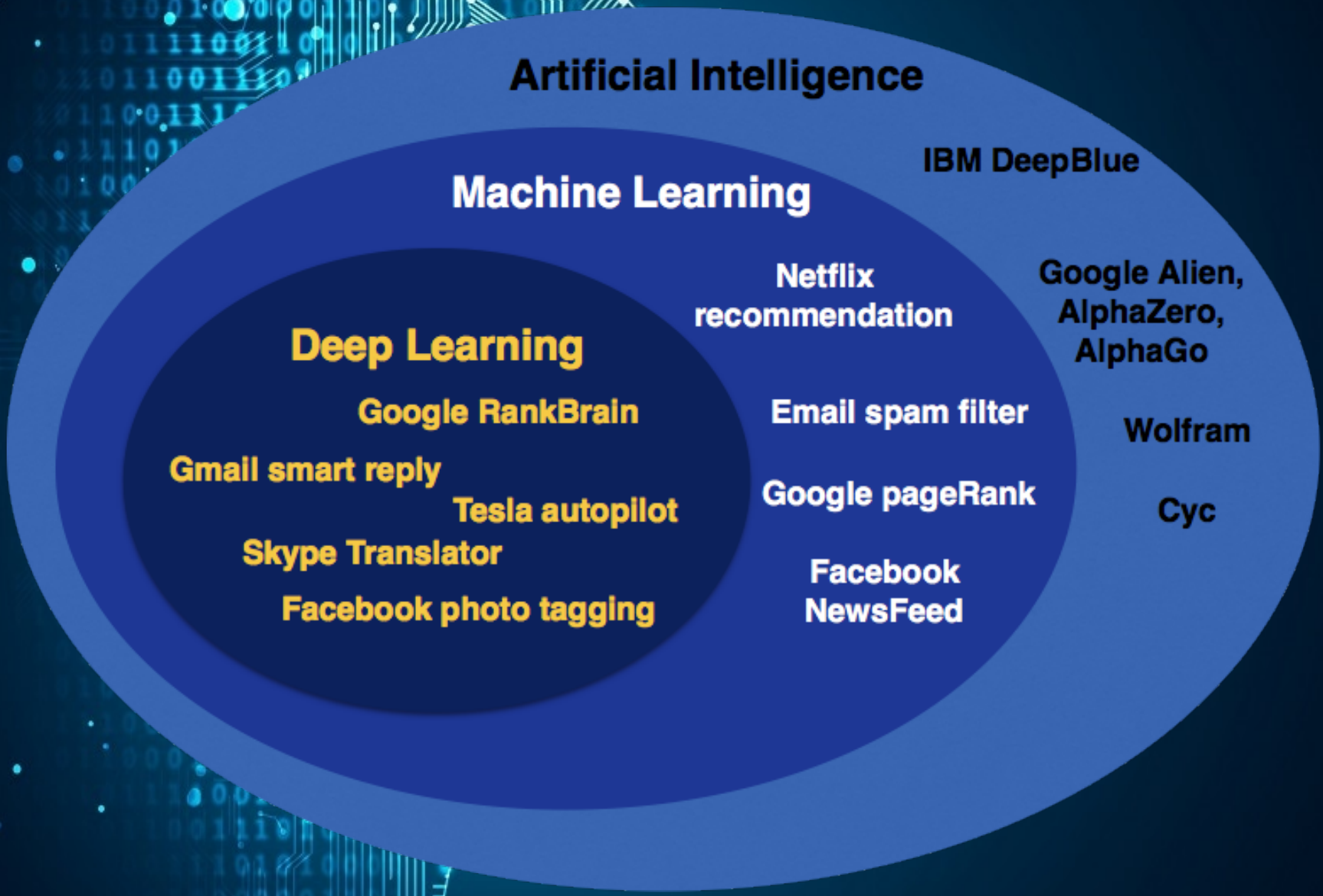


From A. Esteva, B. Kuprel, R. A. Novoa, J. Ko, S. M. Swetter, H. M. Blau & S. Thrun, *Dermatologist-level classification of skin cancer with deep neural networks* in Nature volume 542, pages 115–118 (02 February 2017).

Machine learning today (III)


Artificial Intelligence
Programs able to learn and reason, mimicking human intelligence
Machine Learning
General set of algorithms enabling machines to improve performance when being exposed to more and more data
Deep Learning
Multilayered (deep) Neural Networks
+ vast amount of data

Some achievements



General material (theory)

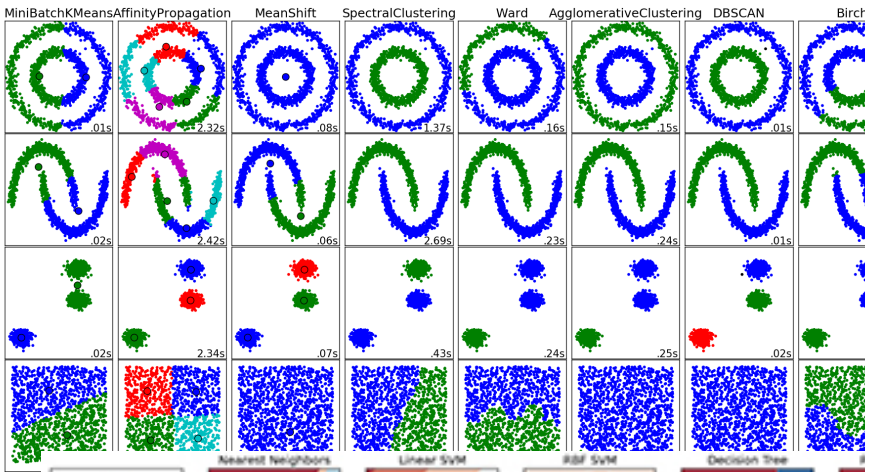
- Reminders in Stats/Probability, Inference.
- Supervised learning
  - Classification
  - Regression
- Unsupervised learning
  - Clustering
  - PCA, ICA
  - Non linear dimensionality reduction
- Directed and undirected graphical models
- Advanced topics (Learning Theory, Adversarial Learning,...)

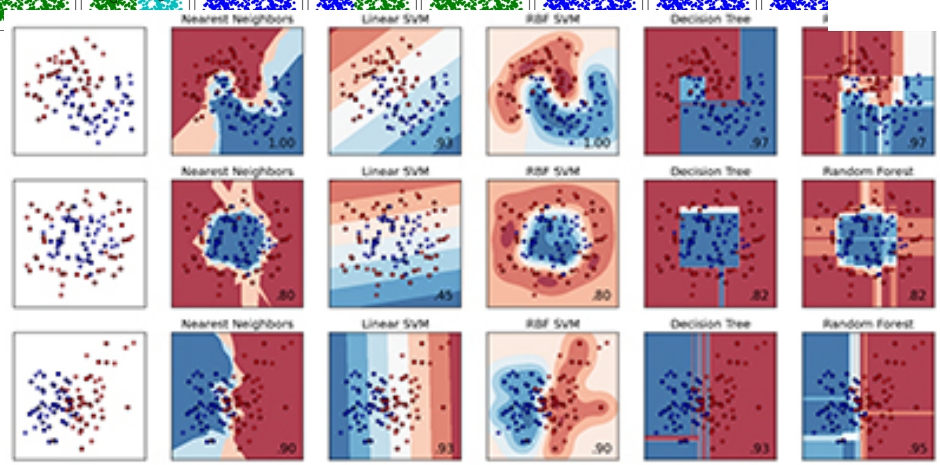
General material (coding)

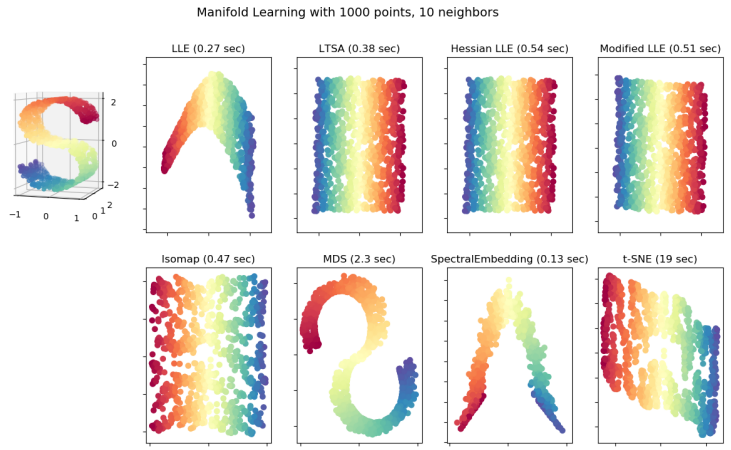
- Programming sessions
  - Scikit-learn, Pandas, MgLearn, PyTorch/TensorFlow
- Personal project
  - Natural Language processing
  - Audio, Video Processing
    - Conversations
    - Faces, Medical diagnosis
    - Sentiments analysis, question answering
  - Time series (financial data, stockmarket)

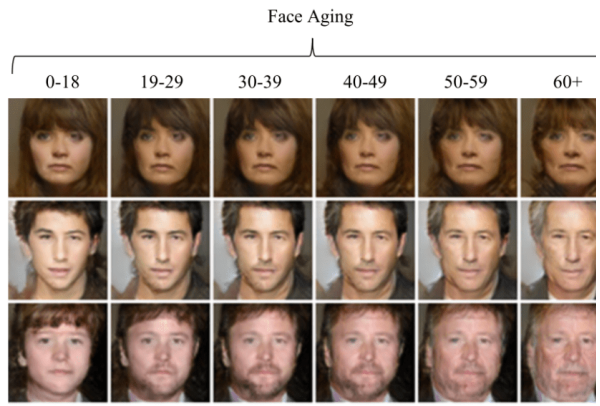
Available tools I

- Programming: Python, Scikit-learn
- Many available datasets and plateforms (register on Kaggle!), also check enigma or Catalyst, quantopian,...



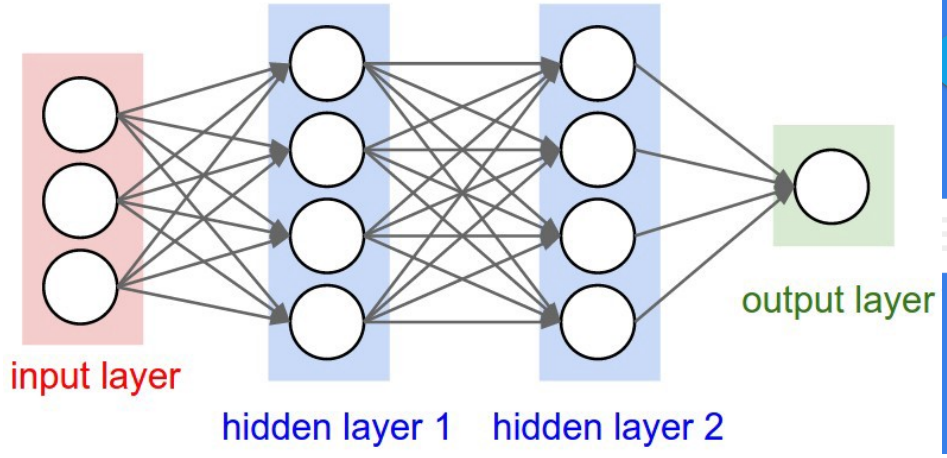


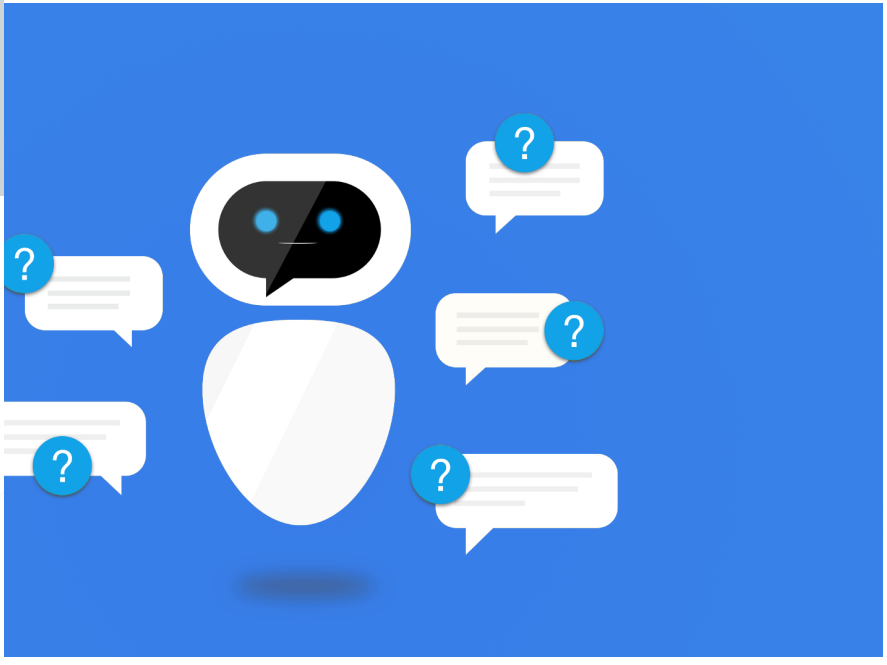




Available tools II

- PyTorch, TensorFlow : Code your own chatbot

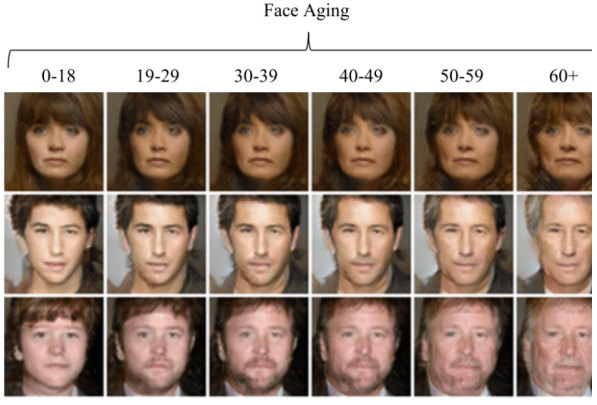




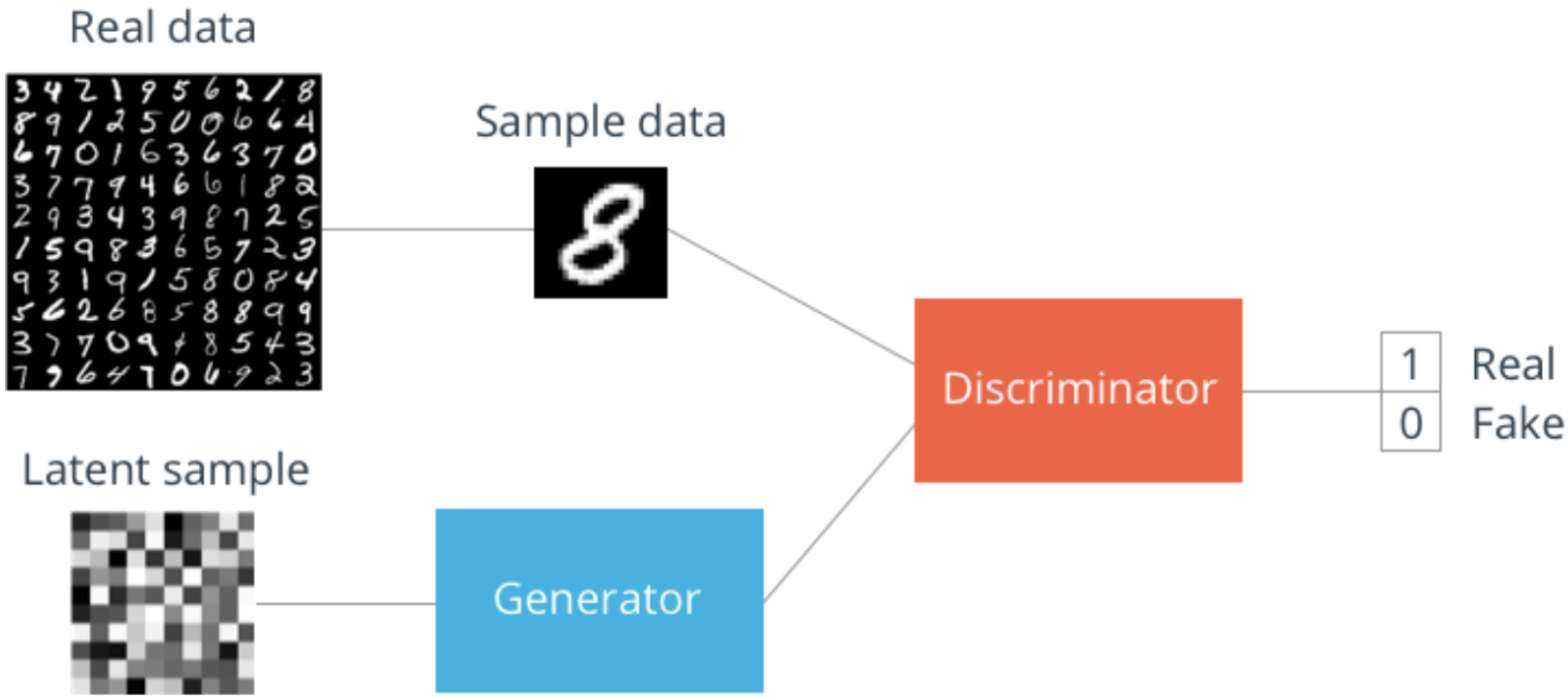
- If time, we will discuss some of the latest models used

Generative Adversarial Networks (GANs): Engine and Applications

How generative adversarial nets are used to make our life better



Generative adversarial networks



Some references

- Website: http://www.augustincosse.com/ml2018

There you can find

- classnotes (coming) + schedule + problem sets.

Additional references (theory side)

- Machine learning: A probabilitic perspective, Murphy,
- Pattern recognition and Machine learning, Bishop,
- The elements of statistical learning, Hastie, Tibshirani, Friedman

Some references (continued)

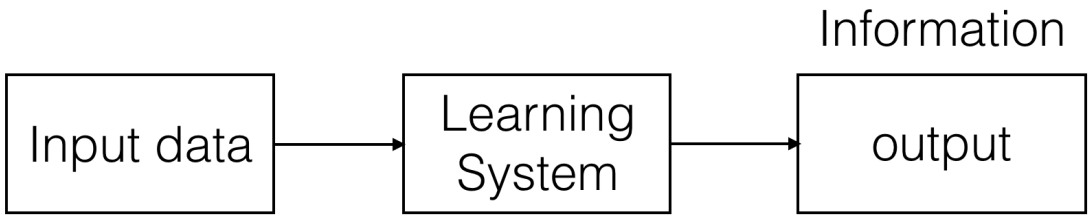
For those interested in startups

My objective: give you the tools, then it's up to you to decide how you want to use them

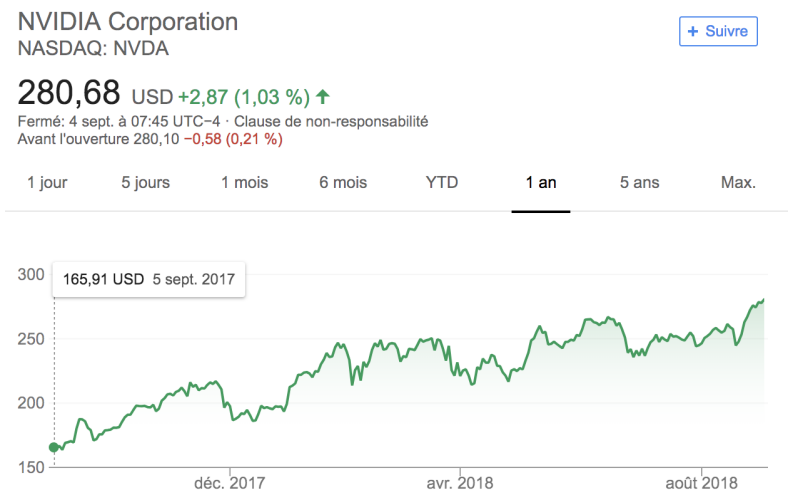
- Many references are available: e.g. Chaos Monkeys (A.G. Martinez), Zero to One (P.Thiel), Creativity inc. (E. Catmull)

Machine Learning is not new..

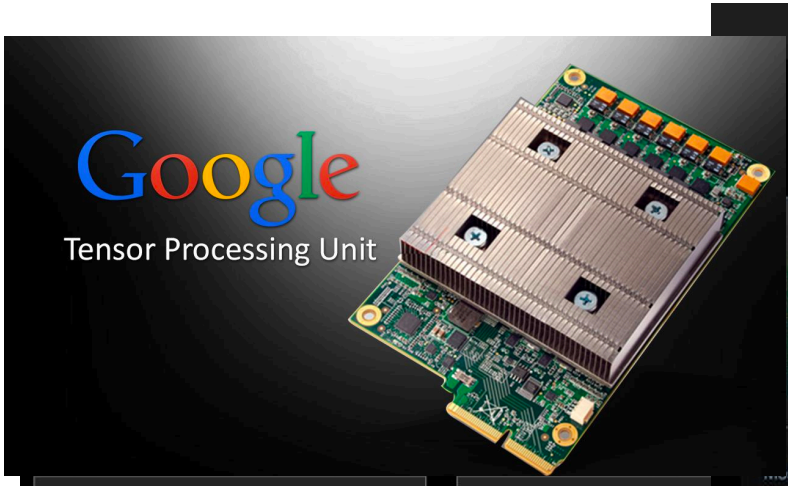
- General principle of machine learning is very simple

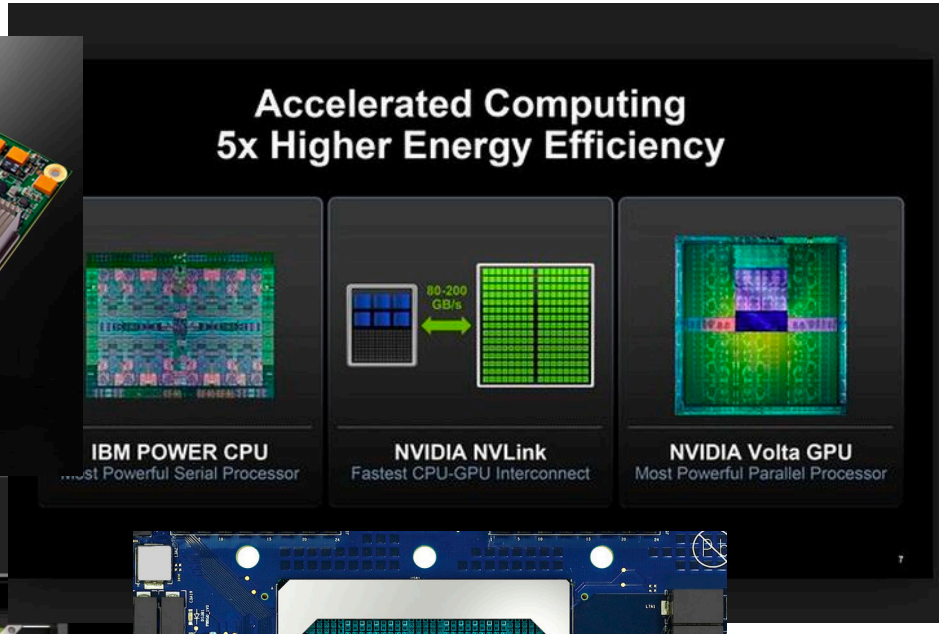


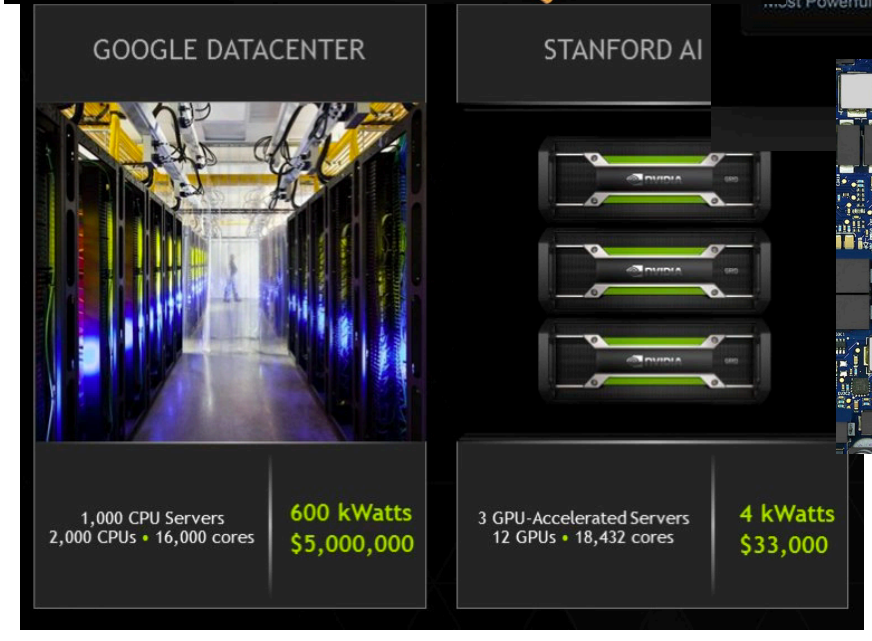
- One of the reason for the renewed excitement is Massive parallelism through Graphical Processing Units $\Rightarrow$ Essential for neural network training on massive databases (think of imageNet, GoogleNet,..)

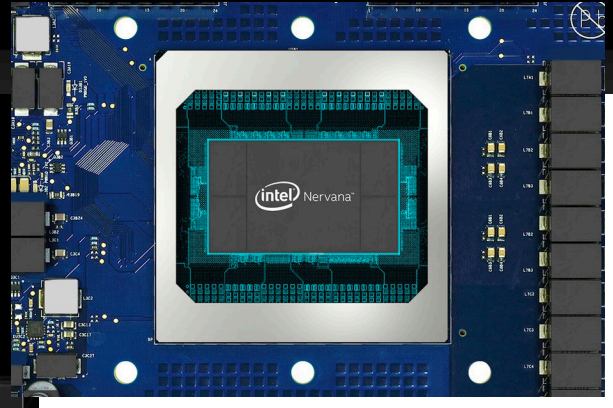


Some new architectures are coming



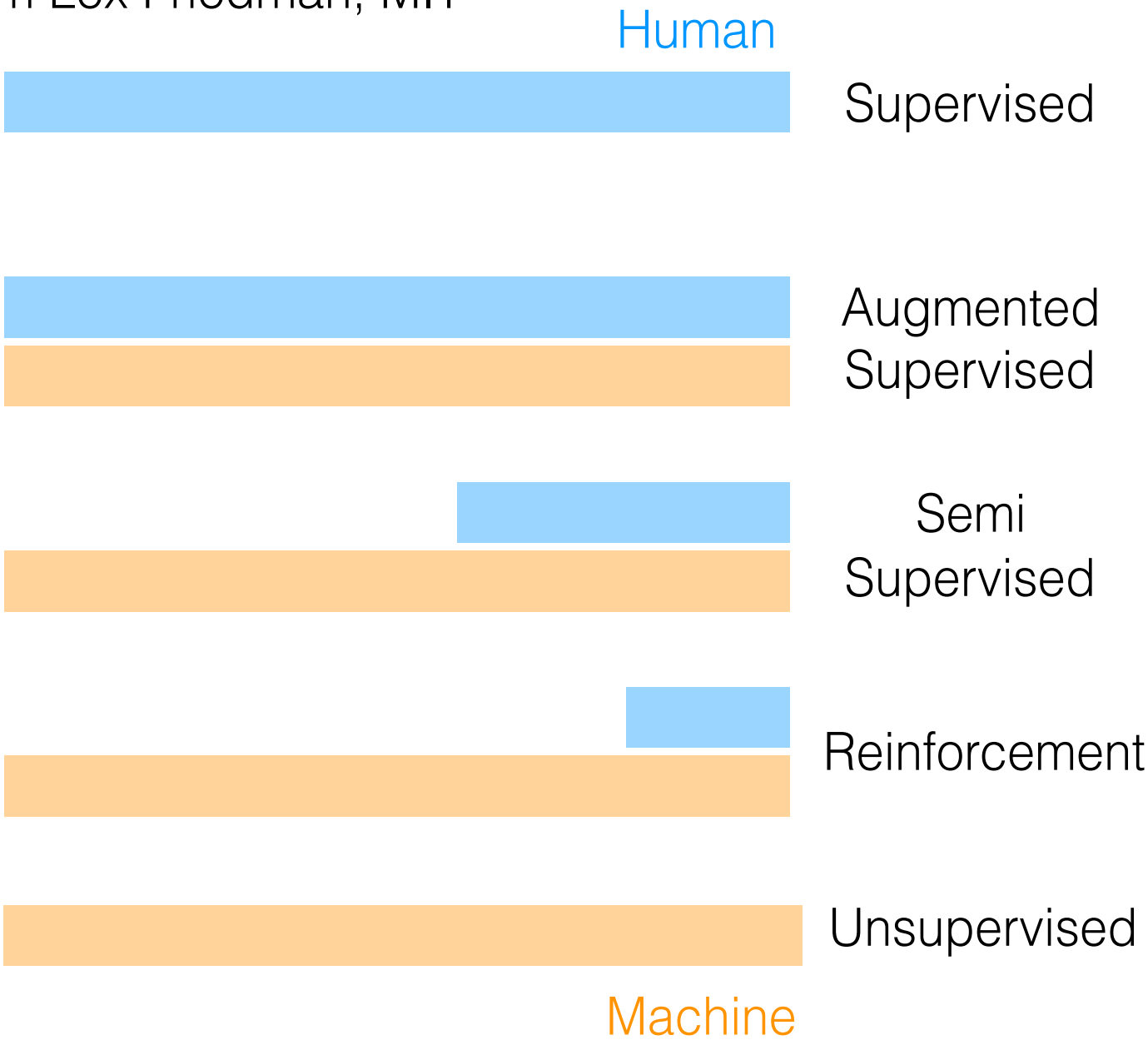






Issue: costly
still Energy greedy

From Lex Friedman, MIT



The big picture: Supervised vs Unsupervised

- Supervised learning tries to understand the relation between data x and the associated labels (knowledge) y based on samples x for which the accompanying labels are known.

  Ex: Handwriting recognition. Data = images from MNIST, labels,knowledge = actual numbers displayed

- Unsupervised learning tries to understand the data without having access to prior knowledge

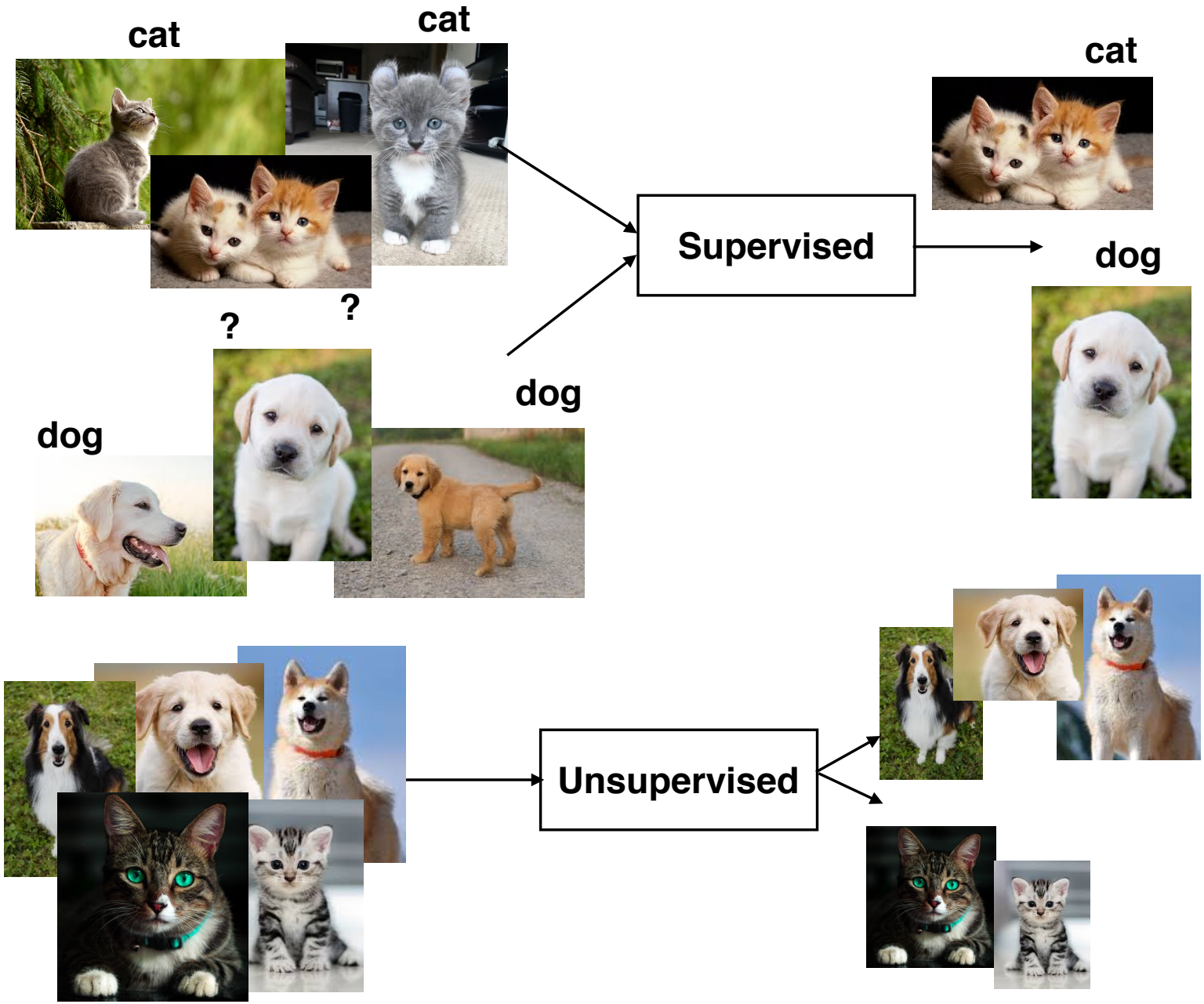
  Ex: customized advertising (cluster users in groups in order to send specific advertising to each group)

The big picture: Augmented Supervised, Semi-supervised and Reinforcement

- Augmented supervised: Human remains at the core but you get the machine to help you with some new classes once in a while. Augmented class = class unknown during training, that appears during test. Once the system can differentiate it from the labeled ones, it will be able to process it later.

- Semi-supervised : small amount of labeled data with large amount of unlabeled data
(labeled and unlabeled data can be used together to learn a manifold)

- Reinforcement: The machine looks for suitable actions, in a given situation, in order to maximize a "reward" (e.g. GANs, Neural Network backgammon playing : board position + dice value $\Rightarrow$ move + reward)


cat
cat
?
?
dog
dog
Supervised
cat
dog
Unsupervised

Supervised learning

Supervised learning is usually split into two categories

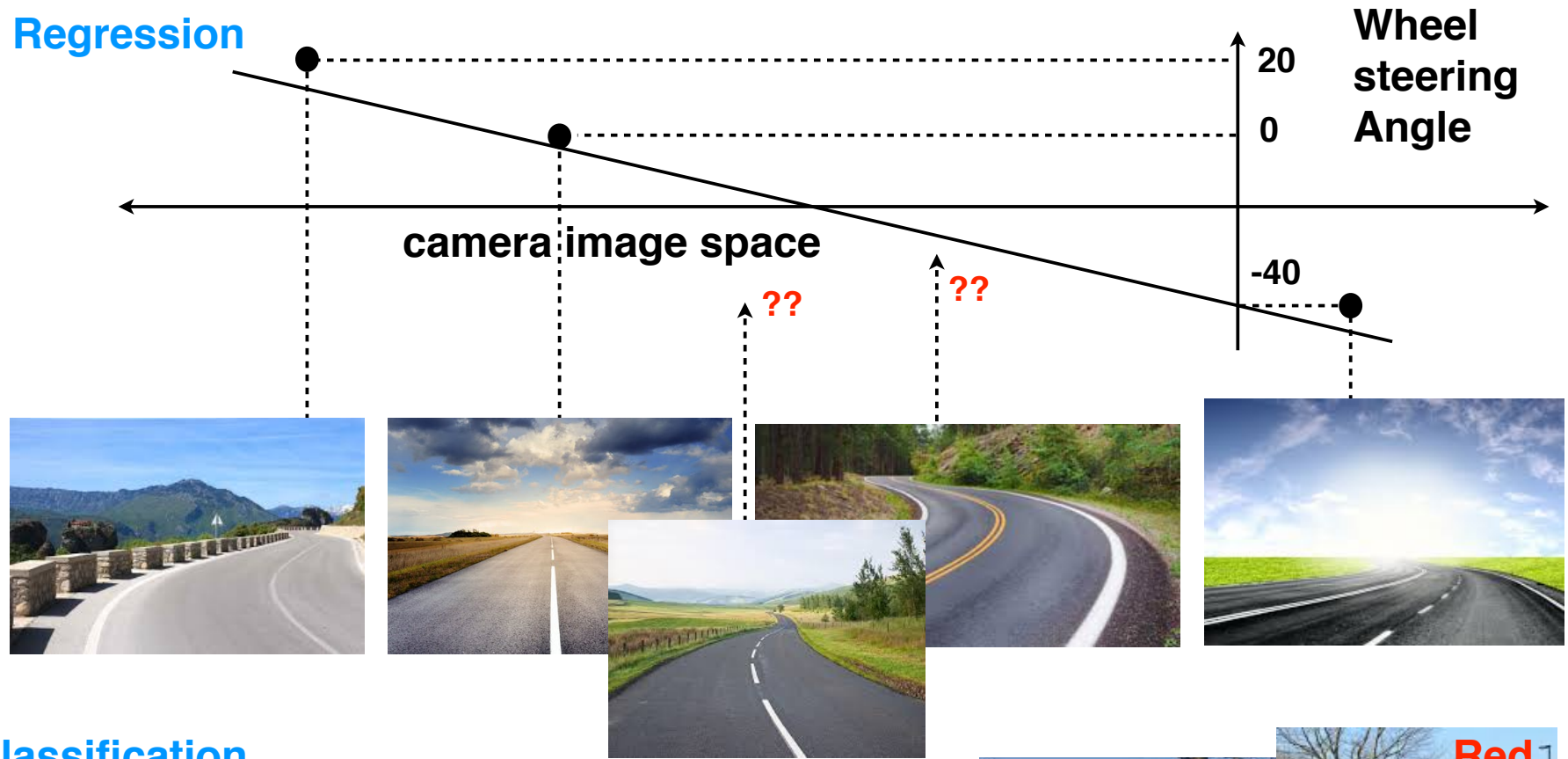
- Regresssion methods

$$Ex. \quad \hat{\beta} = \min_{\beta} \sum_{i=1}^{N} (y_i - \langle x_i, \beta \rangle)^2 \quad \text{(residual SS)}$$

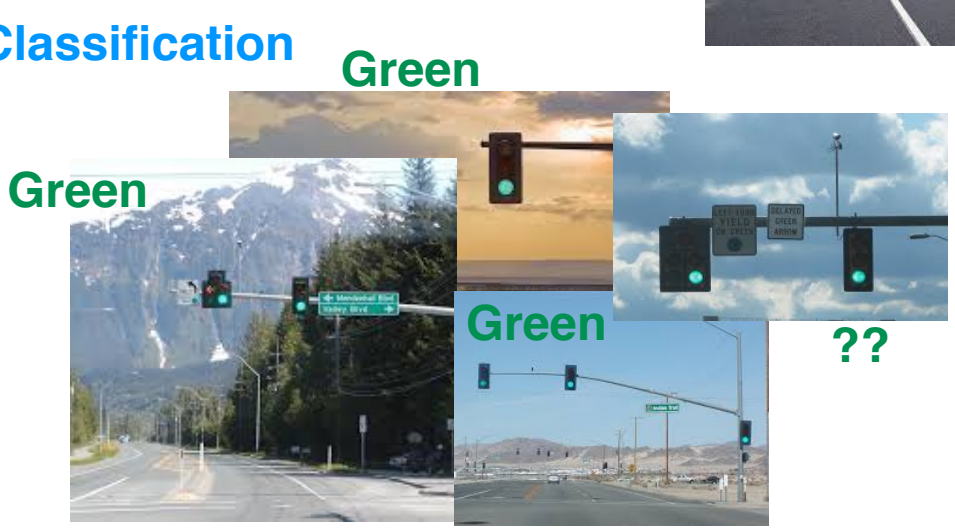
- Classification methods

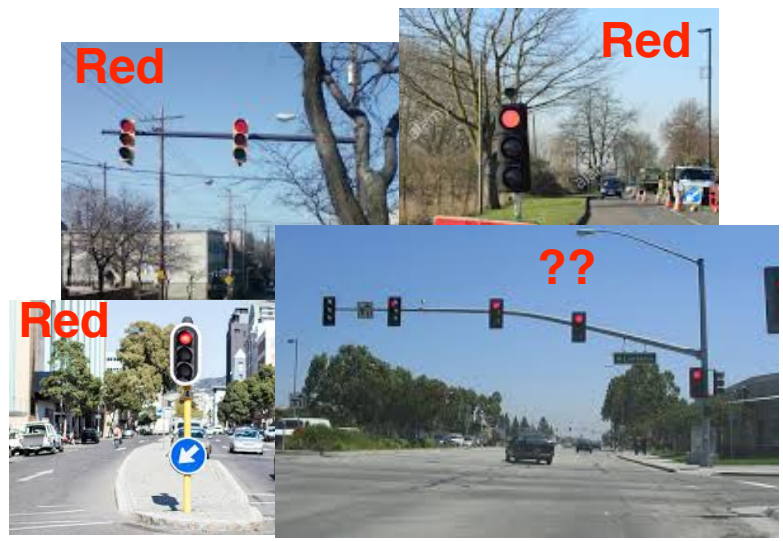
$$Ex. \quad \hat{y}_i = \frac{1}{K} \sum_{k \in \mathcal{N}(x_i)} y(x_k) \quad \text{(KNN)}$$

Regression



Classification





Supervised learning

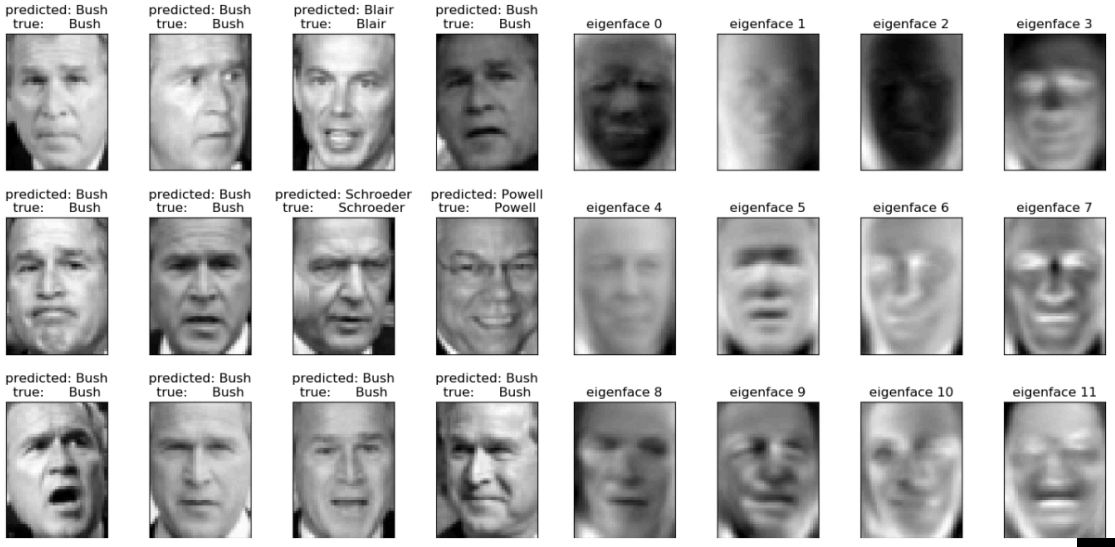
Many possible regression models

- Support vector machines,
- Neural networks
- Kernel methods
- Mixture models

+ Model selection ? Generalization ?

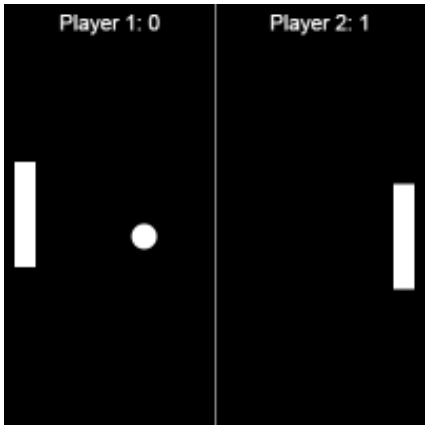
Supervised Learning: What are we going to learn?

SVMs for face recognition



Logistic regression, neural nets for handwritten digits recognition

Neural nets pong training



Parametric vs non-parametric

- Fixed number of parameters = parametric
  $+$: faster to use
  $-$: stronger assumptions regarding data distribution.

  Ex. linear regression

- Number of params grows with training data = non-parametric
  $+$: more flexible regarding data.
  $-$: often computationally intractable for large datasets

  Ex. KNN

Parametric vs non-parametric

- From Hastie, Tishirani, Friedman

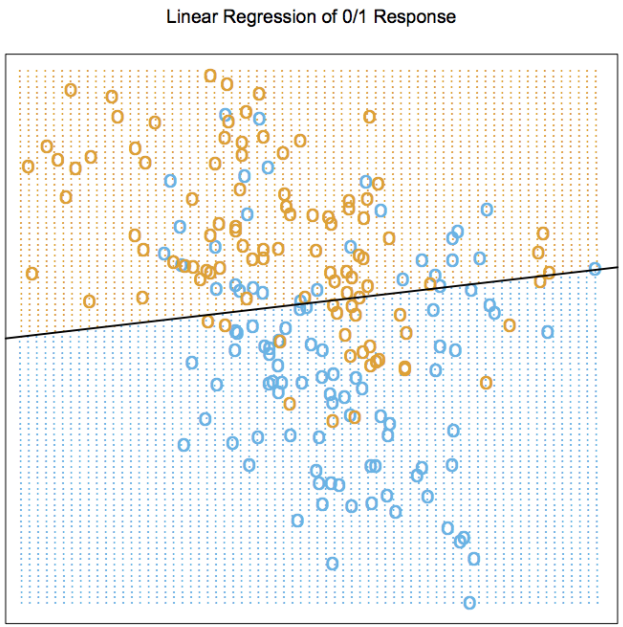


FIGURE 2.1. *A classification example in two dimensions. The classes are as a binary variable (BLUE = 0, ORANGE = 1), and then fit by linear regres The line is the decision boundary defined by $x^T\hat{\beta} = 0.5$. The orange shaded r denotes that part of input space classified as ORANGE, while the blue regi classified as BLUE.*

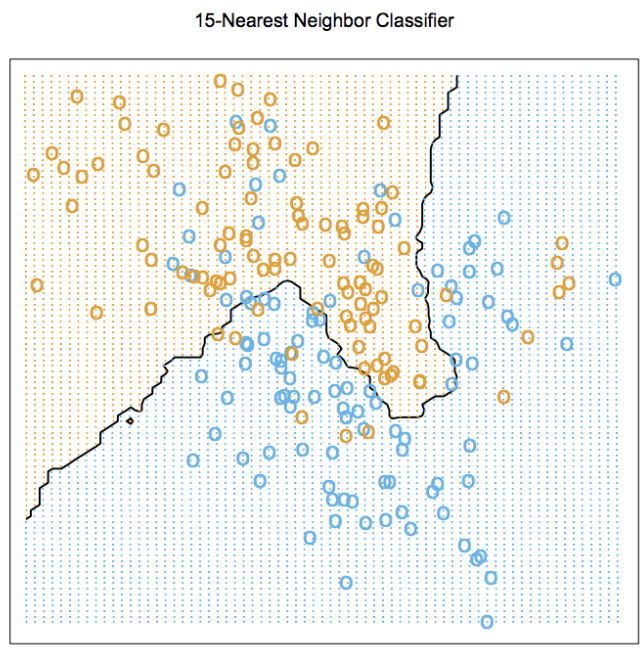


FIGURE 2.2. *The same classification example in two dimensions as in Figure 2.1. The classes are coded as a binary variable (BLUE = 0, ORANGE = 1) and then fit by 15-nearest-neighbor averaging as in (2.8). The predicted class is hence chosen by majority vote amongst the 15-nearest neighbors.*

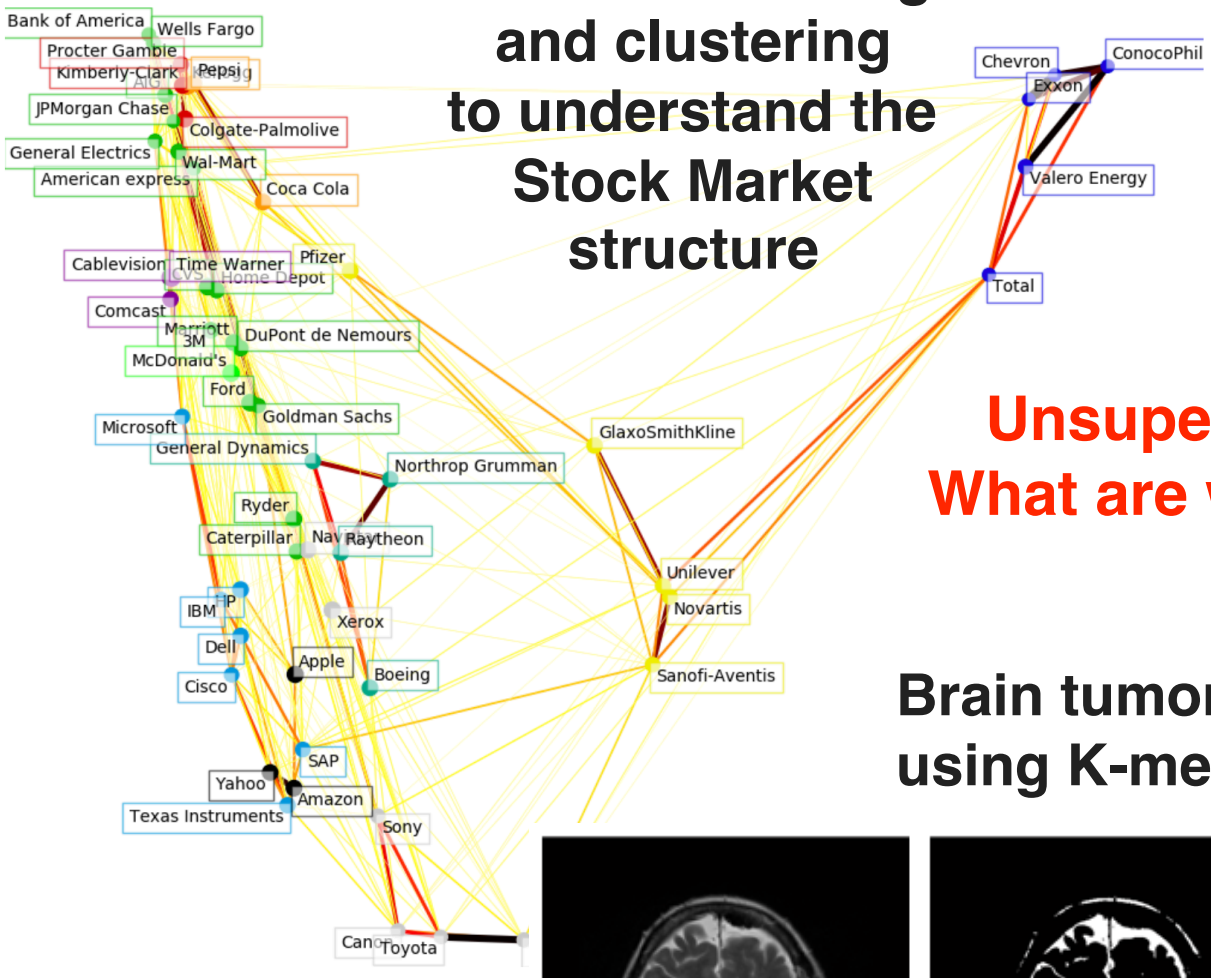
Unsupervised learning

In unsupervised learning, we are only given inputs x_i and we want to extract pattern from the data.

Examples of unsupervised learning approaches include

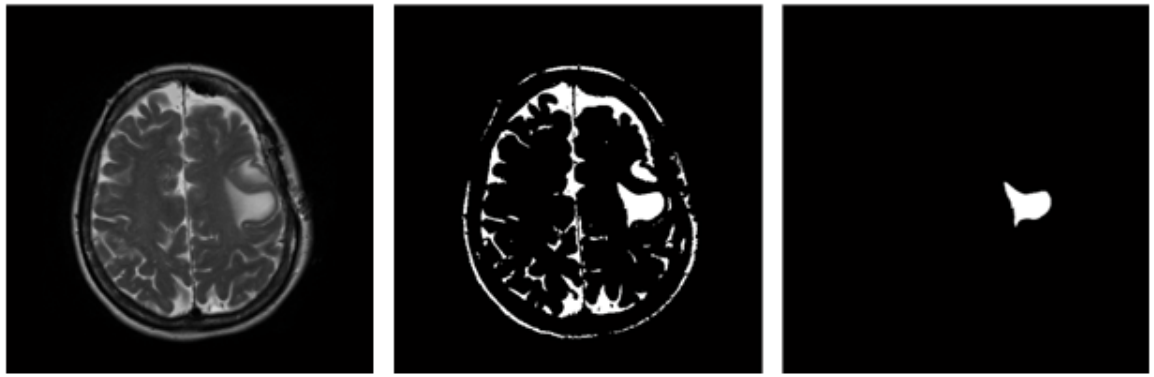
- Clustering
- Self organizing maps
- Principal component analysis
- Non Linear dimensionality reduction

Manifold learning and clustering to understand the Stock Market structure



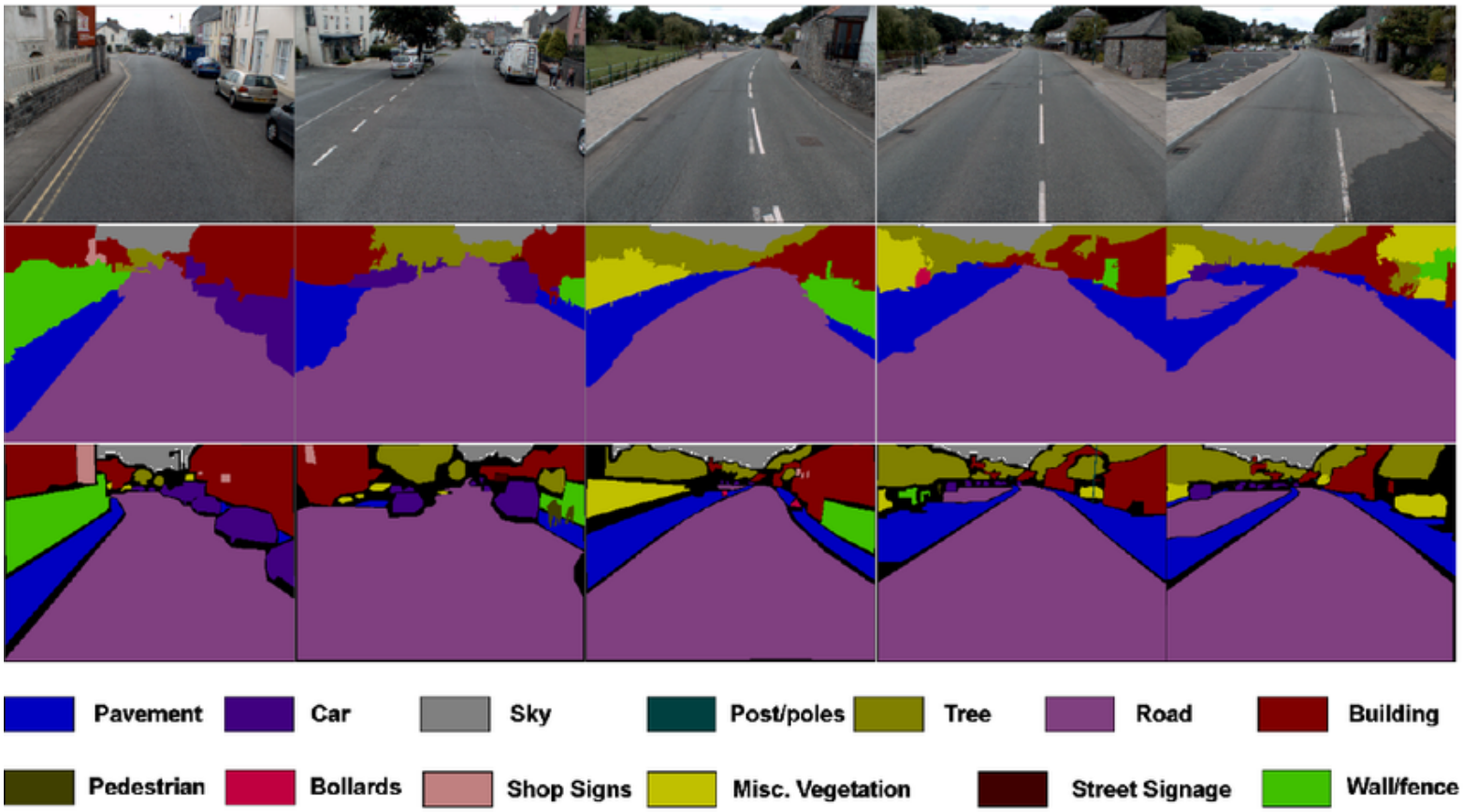
Unsupervised Learning: What are we going to learn?

Brain tumor segmentation using K-means



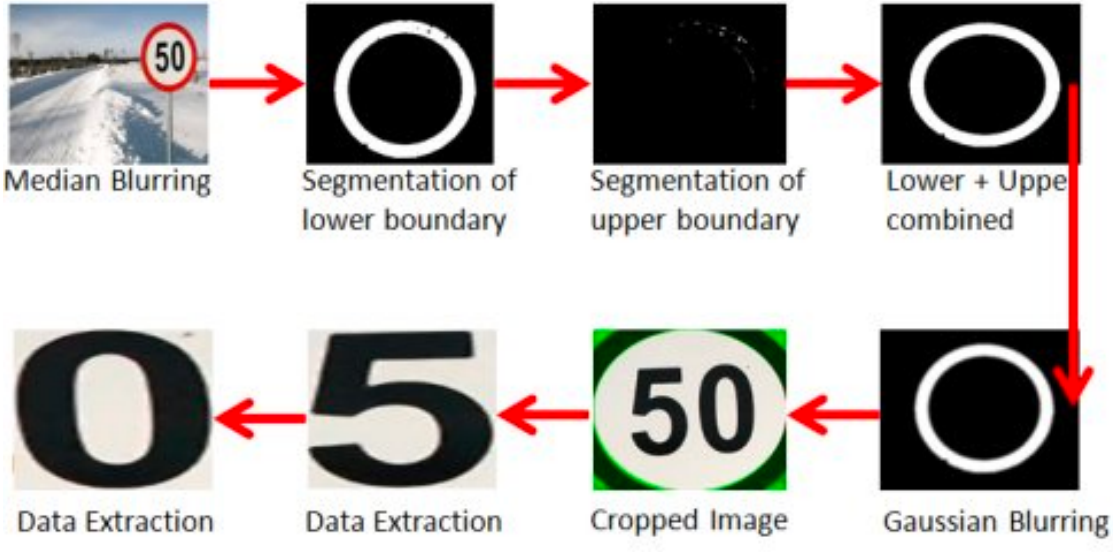
(a) (b) (c)

Image segmentation through clustering



You Can Trick Self-Driving Cars by Defacing Street Signs

By **Catalin Cimpanu**



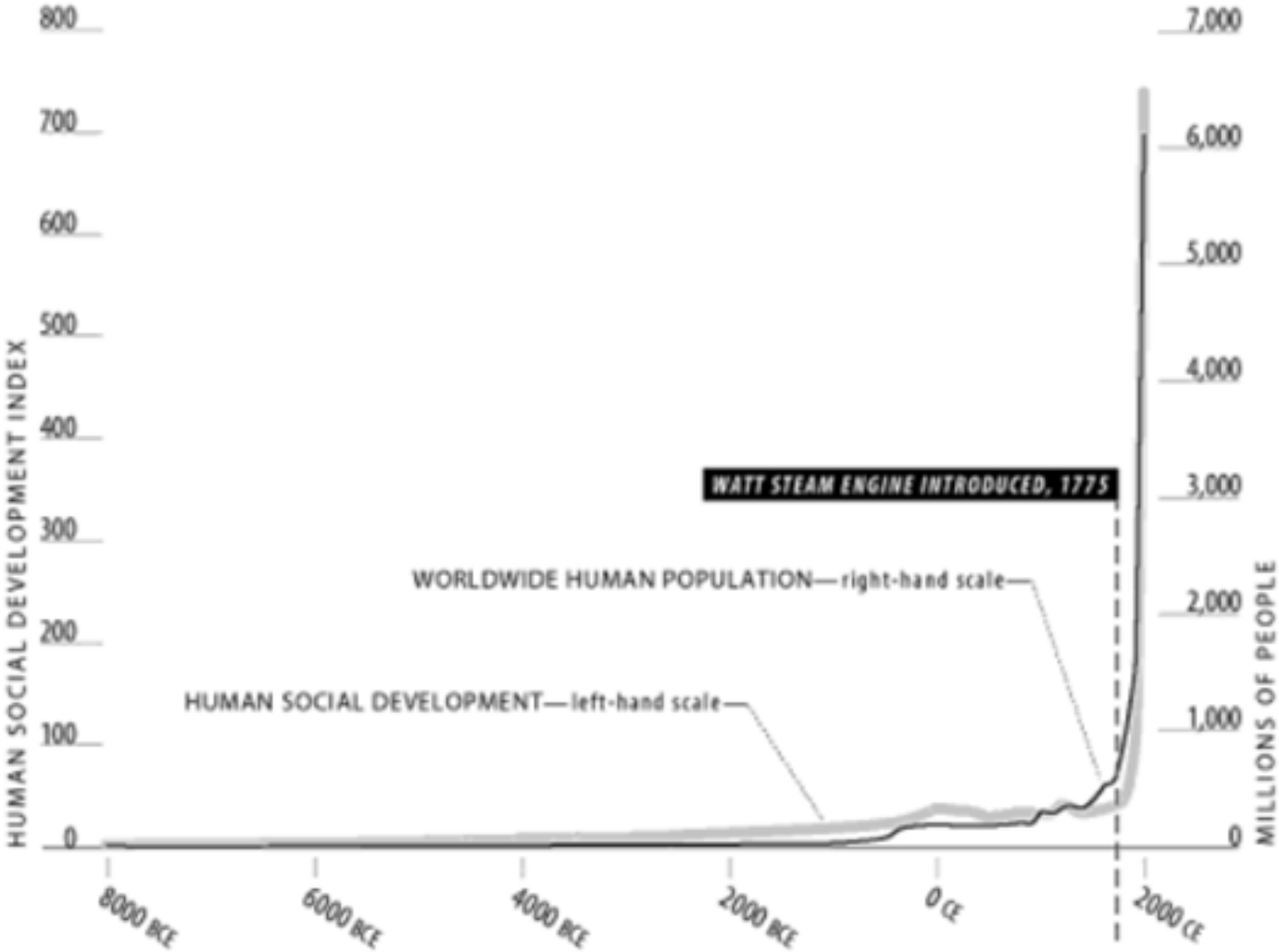
What are we going to learn?
Combined supervised and unsupervised

Samir Bhatt, Bhaskar Trivedi, Ankur Devani, Hemang Bhimani (eInfochips)

Back to the industrial revolution (I)

Erik Brynjolfsson, ICLR 2018

FIGURE 1.2 What Bent the Curve of Human History? The Industrial Revolution.



Back to the industrial revolution (II)

from Erik Brynjolfsson, ICLR 2018

- Why can history tell us?
- Steam engine was classified by Bresnahan et Trajtenberg (1996) as belong to the so-called General Purpose Technologies
- Those technologies are characterized by 3 features:
  - Pervasive
  - Able to be improved over time
  - Able to spawn complemetary innovations
- Does that remind you of something ?

Back to the industrial revolution (II)

from Erik Brynjolfsson, ICLR 2018

- Technology is not neutral
- You can do a small pox vaccine...

Back to the industrial revolution (II)

from Erik Brynjolfsson, ICLR 2018

- Technology is not neutral
- You can do a small pox vaccine...But you can also create a nuclear weapon
- In fact let's compare..

BUSINESS

Is 'Progress' Good for Humanity?

Rethinking the narrative of economic development, with sustainability in mind

JEREMY CARADONNA SEP 9, 2014

Rage against the machine: Luddites smashing a loom. (CHRIS SUNDE / WI

Colin Stretch, general counsel for Facebook, Sean Edgett, acting general counsel for Twitter, Richard Salgado, director of law enforcement and information security at Google, testify before the Senate Judiciary Committee's hearing on 'Extremist Content and Russian Disinformation Online: Working with Tech to Find Solutions' on Capitol Hill in Washington DC on Oct. 31, 2017. Shawn Thew / EPA

Back to the industrial revolution: The 6 Challenges

from Erik Brynjolfsson, ICLR 2018 keynote.

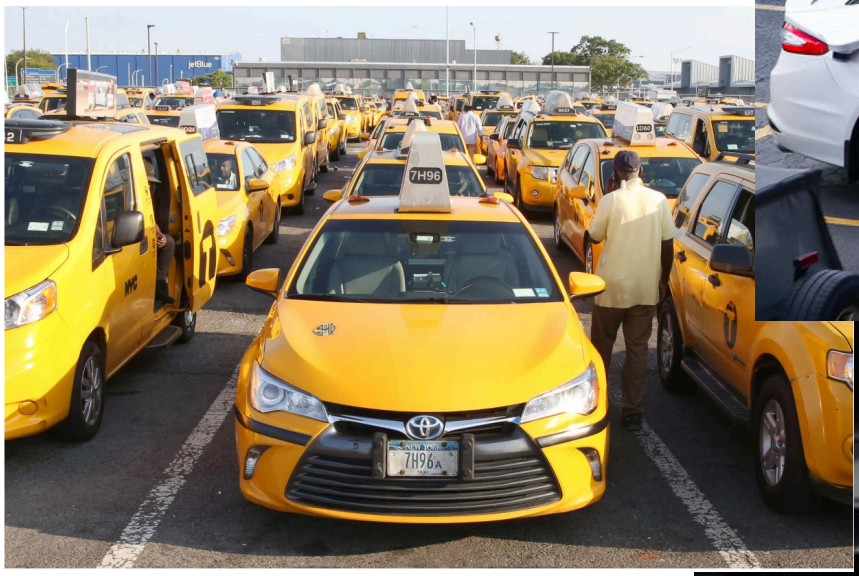
- Economics: Many people left behind
- False news, "cyberbalkanization"
  - Algorithms control what we read, how it is interpreted
  - More facts but more fakes
- Algorithmic bias (Machine can sometimes amplify discriminations when used to hire people, e.g. supervised case)
- End of privacy (IoT and smartphone connection)
- Winner take all markets (further concentration of economic growth)
- Cyber risks (AI bots fighting AI bots, vulnarable voting system)

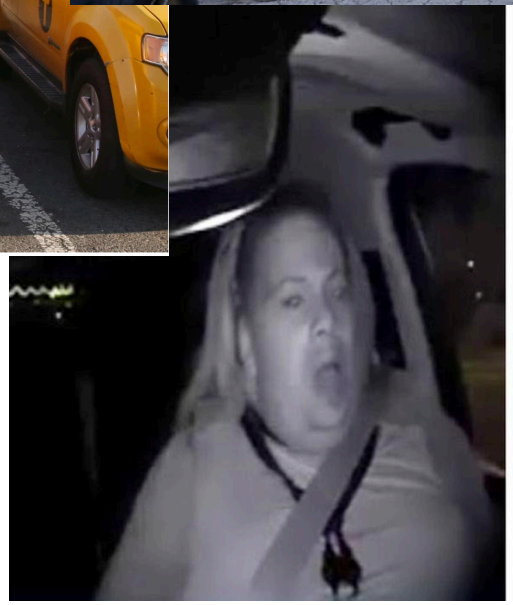
NEW YORK POST

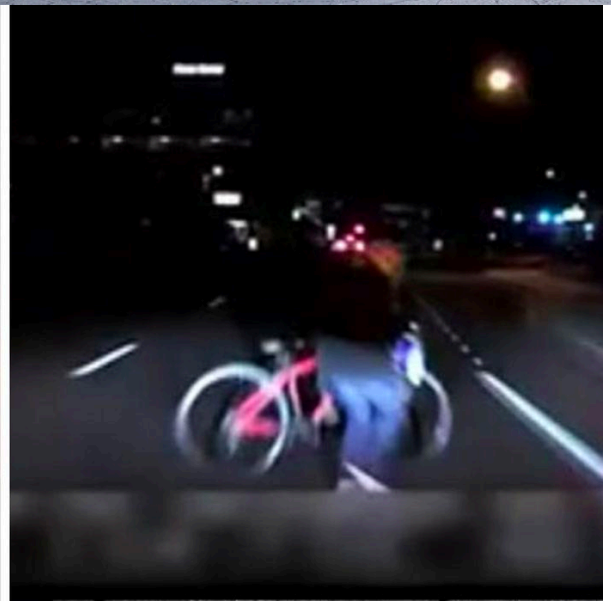
The challenges of driving a yellow cab in the age of Uber

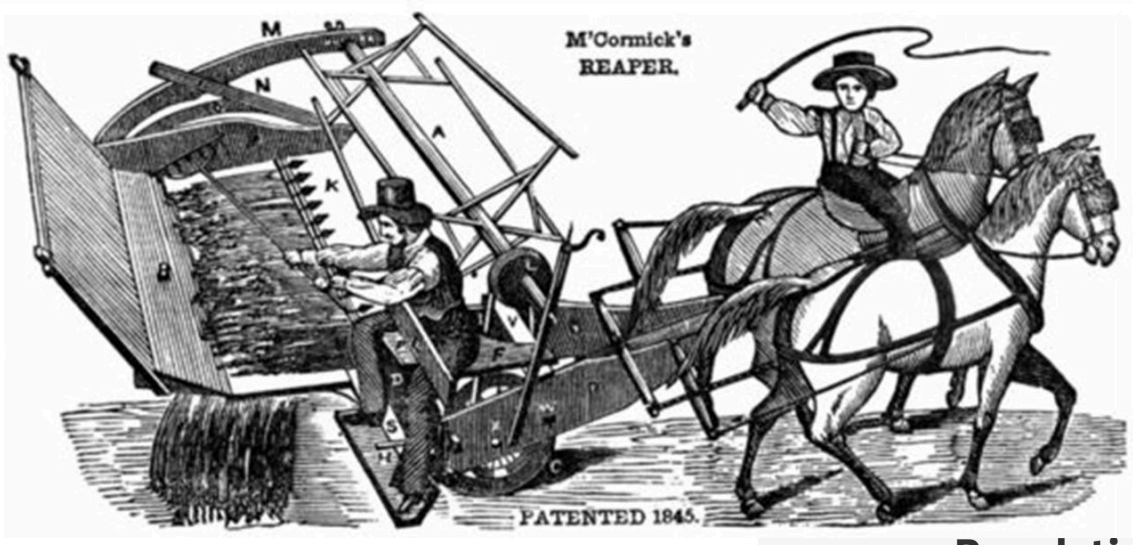
By John Crudele

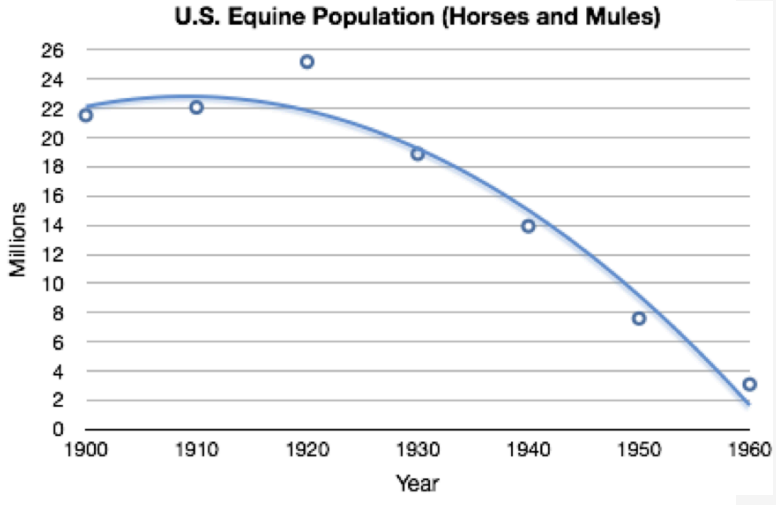
September 18, 2017 | 10:35pm



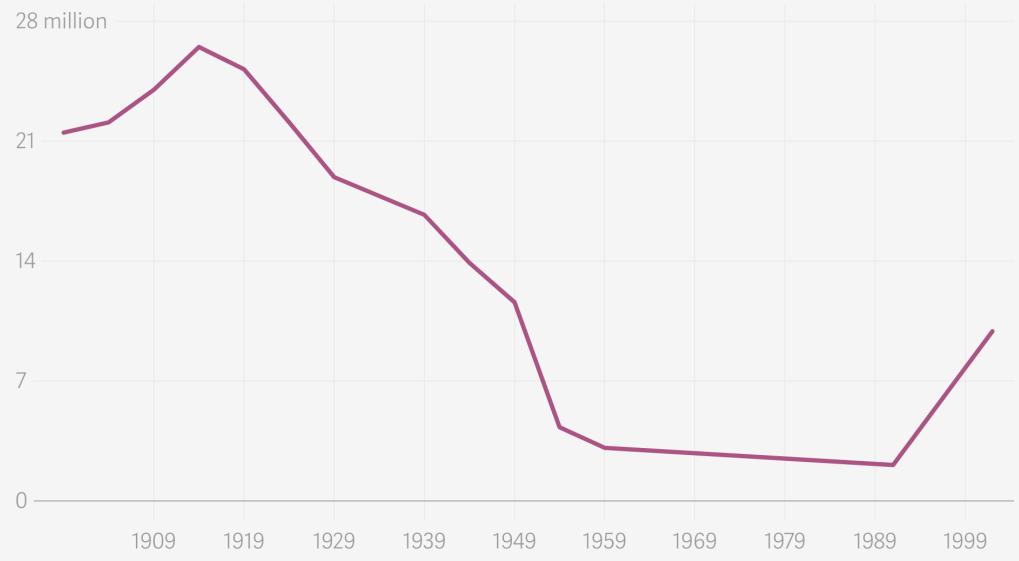








Population of horses in the US during industrialization



CIO JOURNAL.

What Machine Learning Can and Cannot Do

Jul 27, 2018 1:56 pm ET

A doctor examines a magnetic resonance image of a human brain during a Beijing neuroimaging competition between human doctors and AI, June 30, 2018. **PHOTO:** MARK SCHIEFELBEIN / ASSOCIATED PRESS

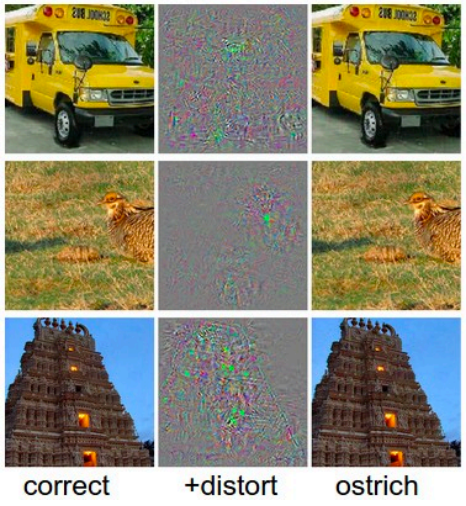
What Machine Learning can and cannot do

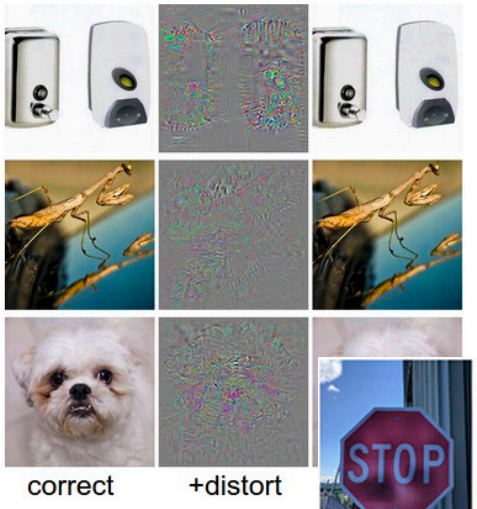
- We have seen many achievements (essentially in vision, language)..
- In particular Supervised learning has been quite successful
- But there are still plenty of tasks that computers still cannot handle (see Lex Friedman, MIT Sloan lecture)
  - Awareness of self
  - Emotion
  - Imagination
  - Morality
  - Consciousness
  - high level reasoning

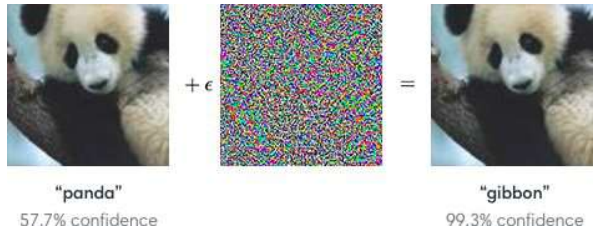
Still many tasks that machines cannot do

from Erik Brynjolfsson, ICLR 2018 keynote.

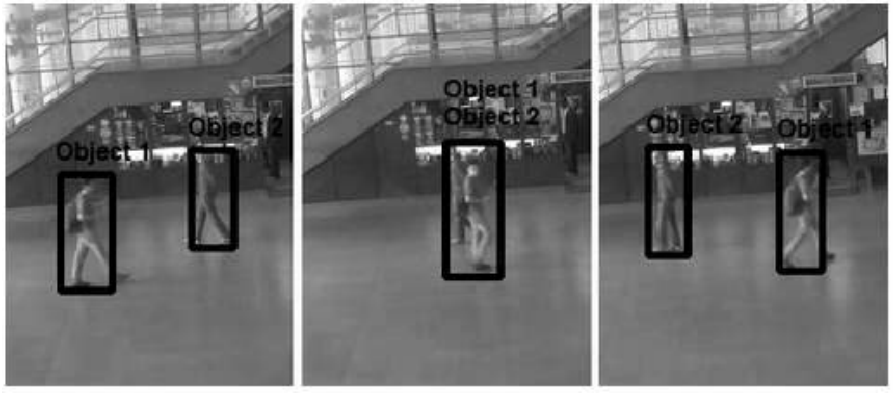
Immediate Challenges

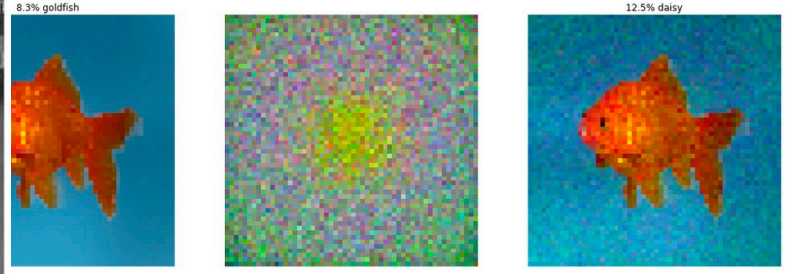






Neural Nets and machine vision can be fooled by very small distortions





Immediate Challenges

(Lex Friedman, MIT Sloan)

- Occlusions
- Sensor spoofing (camera, Lidar)
- adversarial noise
- Risk quantification
- Data is costly $\Rightarrow$ Ideally, we would like to move to

Statistics and probability

- Why using stats/proba?
- Machine Learning relies on complex distributions (cancerous cells, possible moves in Go, Existing sign roads, possible evolutions of stocks,...)
- Only a few samples are usually available
- $\Rightarrow$ We need a way to measure how well those samples are representing the underlying (unknown) distribution

Why is that important?

Self-Driving Uber Car Kills Pedestrian in Arizona, Where Robots Roam

A woman crossing Mill Avenue at its intersection with Curry Road in Tempe, Ariz. on Monday. A pedestrian was struck and killed by a self-driving Uber vehicle at the intersection a night earlier.
Caitlin O'Hara for The New York Times

Reminders (I)

(Discrete sets of events)

- Sum rule $p(X) = \sum_Y p(X|Y)$
- Product rule $p(X, Y) = p(X|Y)p(Y)$
- Bayes theorem

$$p(Y|X) = \frac{p(X|Y)p(Y)}{p(X)}$$

(continuous sets of events)

- density $p(x)$,

$$p(x \in [a, b]) = \int_a^b p(x)dx, \quad p(x) = \int p(x, y)dy$$

Reminders (II)

- Cumulative distribution Function (CDF) $F(z) = \int_{-\infty}^{z} p(x)\, dx$
- Expectation $\mathbb{E}[x] = \int x p(x) dx$, $\mathbb{E}[x] = \sum_i x_i p(x_i)$
- Conditional expectation $\mathbb{E}_x f(x|y) = \sum_x f(x) p(x|y)$
- Variance $\mathrm{Var}[x] \equiv \mathbb{E}\left\{(x - \mathbb{E}x)^2\right\}$
- Covariance $\mathrm{Cov}[x, y] \equiv \mathbb{E}\left\{(x - \mathbb{E}x)(y - \mathbb{E}y)\right\}$

Reminders (III) A few important distributions

- The gaussian distribution

$$\mathcal{N}(x|\mu,\sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{1}{2\sigma^2}(x-\mu)^2\right)$$

- Uniform distribution: $P(y) = \frac{1}{|b-a|}, \quad y \in [a,b]$

- χ^2 distribution: $\chi^2 \sim \sum_{i=1}^{N} {Z_i}^2$ with Z_i independent standard normal RV.

Reminders (IV) A few important distributions

- Binary variables: Bernoulli and Rademacher,

$$\text{Bern}(x|\mu) = \mu^x(1-\mu)^{1-x}, \quad x = \begin{cases} 1 \\ 0 \end{cases}, 0 \leq \mu \leq 1$$

$$\text{Rademacher:} \quad \varepsilon(x) = \begin{cases} (1/2), & x = +1 \\ (1/2), & x = -1 \\ 0, & \text{otherwise} \end{cases}$$

The exponential family

- Many of the distributions we have discussed are part of a general family called The exponential family
- The exponential family has many interesting properties
  - It is the only family of distribution with finite-sized sufficient statistics (see next slides)
  - It is the only family with known conjugate priors
  - It is at the core of generalized linear models
  - it is at the core of variational inference
- we will come back to these notions later

The exponential family

- A pdf $p(\boldsymbol{x}|\theta)$ is said to be in the exponential family for $\boldsymbol{x} = (x_1, \ldots, x_m)$ and $\theta \subseteq \mathbb{R}^d$ if

$$\begin{aligned} p(\boldsymbol{x}|\boldsymbol{\theta}) &= \frac{1}{Z(\boldsymbol{\theta})} h(\boldsymbol{x}) \exp(\theta^T \phi(\boldsymbol{x}) \\ &= h(\boldsymbol{x}) \exp(\boldsymbol{\theta}^T \phi(\boldsymbol{x}) - A(\boldsymbol{\theta})) \end{aligned}$$

- $Z(\theta)$ and $A(\theta)$ are defined as

$$\begin{aligned} Z(\theta) &= \int_{\mathcal{X}^m} h(\boldsymbol{x}) \exp[\theta^T \phi(x)] \, dx \\ A(\theta) &= \log(Z(\theta)) \end{aligned}$$

- $Z(\theta)$ is called the partition function, θ are the mutual parameters, $\phi(x) \in \mathbb{R}^d$ is the vector of sufficient statistics, $A(\theta)$ is the log partition function or cumulant function.

The exponential family

- Two examples
  - Bernoulli

$$\begin{aligned}\text{Ber}(x|\mu) &= \mu^x(1-\mu)^{1-x} = \exp(x\log(\mu) + (1-x)\log(1-\mu)) \\ &= \exp(\phi(x)^T\theta)\end{aligned}$$

  - Univariate Gaussian
- The Uniform distribution does not belong to the exponential family

Parameter/model inference: Bayesian vs frequentist

- The linear regression model is a special instance of a more general idea called model inference (among which one finds the MLE)
- We will study the notion of inference in more details later in the class. For now we only cover the main ideas.
- Inference can be used in both supervised (learn new labels from training labels) and unsupervised (learn parameters from distribution) frameworks
- You will often hear about frequentist vs Bayesian approaches.

Parameter/model inference: Bayesian vs frequentist

- Bayesian statistics.
  - Considers the (distribution) parameters as random
  - Relies heavily on the posterior distribution $p(\boldsymbol{\theta}|\mathcal{D})$
  - dominated statistical practice before 20^{th} century
  - Ex: MAP $\underset{\theta}{\text{argmax}}\, P(\mathcal{D}|\theta)P(\theta)$
- Frequentist statistics (a.k.a classical stat.)
  - Parameters $\boldsymbol{\theta}$ viewed as fixed, sample $\mathcal{D}$ as random (Randomness in the data affects the posterior)
  - Relies on the likelihood or some other function of the data
  - dominated statistical practice during 20^{th} century
  - Ex. MLE : $\underset{\theta}{\text{argmax}}\, P(\mathcal{D}|\theta)$

Bayesian statistics: Some vocabulary

- We saw Bayesian inference relies on the posterior $p(\theta|\mathcal{D})$
- The posterior reads from the Bayes rule as

$$p(\theta|\mathcal{D}) = \frac{p(\mathcal{D}|\theta)p(\theta)}{p(\mathcal{D})} = \frac{p(\mathcal{D}|\theta)p(\theta)}{\int p(\mathcal{D}|\theta)p(\theta)d\theta}$$

- $p(\theta)$ is called the prior, $p(\mathcal{D}|\theta)$ is called the likelihood function and $Z = p(\mathcal{D})$ is the normalizing constant (independent of θ)
- Given a set of patterns $(\boldsymbol{x}_\mu, \boldsymbol{y}_\mu)$, classifiers are usually of two types:
  - Generative (learn model for $p(\boldsymbol{x}, \boldsymbol{y}|\theta)$)
  - Discriminative (learn model for $p(\boldsymbol{y}|\boldsymbol{x}, \theta)$)

Bayesian statistics: Some vocabulary

- An example of discriminative classifier : Logistic regression
  - Here we take $\mu(\boldsymbol{x}) = \text{sigm}(\boldsymbol{w}^T \boldsymbol{x})$ and define the classifier as a Bernoulli distribution

$$p(y|\boldsymbol{x}, \boldsymbol{w}) = \text{Ber}(y|\mu(\boldsymbol{x}))$$

  - Good when the output is binary

- An example of generative classifier :
  - relies on the assumption that the features (hidden variables) are independent

$$p(\boldsymbol{x}|y = c, \boldsymbol{\theta}) = \prod_{j=1}^{D} p(x_j|y = c, \theta_{jc})$$

  - $\theta_{j,c}$ is the parameters of the distribution of class c for j^{th} entry in the D-dimensional pattern vector $\boldsymbol{x} \in \{1, \ldots, K\}^D$.

- We will study those models in further detail when discussing classifiers.

Bayesian statistics

- In Bayesian statistics, randomness is most often used to encode uncertainty
- The posterior $p(\boldsymbol{\theta}|\mathcal{D})$ summarizes all we know on the parameters
- Bayesian inference is not always the right choice because of the following
  - The Mode is not a typical point in the distribution
  - MAP estimator depends on re-parametrization

Bayesian statistics: Drawbacks and solutions

- A solution to the first part is to use a more robust loss function $\ell(\hat{\theta}, \theta)$
- A solution to the second part is to replace the MAP with the following estimator (when available)

$$\hat{\theta} = \underset{\theta}{\text{argmax}} p(\mathcal{D}|\boldsymbol{\theta}) p(\boldsymbol{\theta}) |\boldsymbol{I}(\boldsymbol{\theta})|^{-1/2} \tag{1}$$

where $\boldsymbol{I}(\boldsymbol{\theta})$ is the Fischer information matrix

Fischer information matrix

- For a generative model $p(\boldsymbol{x}|\theta)$, we let $g(\theta, \boldsymbol{x})$ denote the Fischer score

$$g(\theta, \boldsymbol{x}) = \nabla_\theta \log(p(\boldsymbol{x}|\theta))$$

- the Fischer Kernel is the defined as

$$k(\boldsymbol{x}, \boldsymbol{x}') = g(\boldsymbol{\theta}, \boldsymbol{x})^T \boldsymbol{F}^{-1} g(\boldsymbol{\theta}, \boldsymbol{x}')$$

- The matrix $\boldsymbol{F}$ is called the Fischer matrix and defined as

$$\boldsymbol{F} = \mathbb{E}_{\boldsymbol{x}} \left\{ g(\boldsymbol{\theta}, \boldsymbol{x}) g(\boldsymbol{\theta}, \boldsymbol{x})^T \right\}$$

- Note that it is often computed empirically as

$$\boldsymbol{F} \approx \frac{1}{N} \sum_{n=1}^{N} g(\theta, \boldsymbol{x}) g(\boldsymbol{\theta}, \boldsymbol{x})^T$$

Occam's razor and Model selection

- Only looking for the best model often leads to overfitting (we will see that later in more details)
- Bayesian framework offers and alternative called Bayesian model selection
- For a family of models, we can express the posterior

$$p(m|\mathcal{D}) = \frac{p(\mathcal{D}|m)p(m)}{\sum_{m \in \mathcal{M}} p(m|\mathcal{D})} \propto p(\mathcal{D}|m)p(m)$$

where $p(\mathcal{D}|m) = \int p(\mathcal{D}|\boldsymbol{\theta})p(\boldsymbol{\theta}|m)d\boldsymbol{\theta}$ is called the marginal likelihood, integrated likelihood or evidence

Occam's razor

- Integrating the parameters $\boldsymbol{\theta}$ such as in

$$p(\mathcal{D}|m) = \int p(\mathcal{D}|\boldsymbol{\theta})p(\boldsymbol{\theta}|m)d\boldsymbol{\theta}$$

  acts as a natural regularization and prevents overfitting when solving for $\max_m p(m|\mathcal{D})$. This idea is known as Bayesian Occam's razor

- The evidence $p(\mathcal{D}|m)$ can be understood as the probability to generate a particular dataset from a family of model (all values of the parameters included).

- When the family of models is too simple, or too complex, this probability will be low.

Bayesian decision theory

- How do we resolve the lack of robustness of Bayesian estimators vis a vis the distribution (recall the bimodal distribution)?
- Statistical decision theory can be viewed as a game against nature.
- Nature has a parameter value in mind and gives us a sample
- We then have to guess what the value of the parameter is by choosing an action a
- As an additional piece of information, we also get a feedback from a loss function $L(y, a)$ which tells us how compatible our action is w.r.t Nature's hidden state.

Bayesian decision theory

- The goal of the game is to determine the optimal decision procedure,

$$\underset{a \in \mathcal{A}}{\operatorname{argmin}} \, \mathbb{E}\left\{L(y, a)\right\}$$

- In economics $L(y, a) = U(y, a)$ and leads to the Maximum utility principle which is considered as rational behavior

$$\delta(x) = \underset{a \in \mathcal{A}}{\operatorname{argmax}} \, \mathbb{E}\left\{U(y, a)\right\}$$

- In the Bayesian framework, we want to minimize the loss over the models compatible with the observations $\{\boldsymbol{x}_\mu\}$

$$\delta(x) = \underset{a \in \mathcal{A}}{\operatorname{argmin}} \mathbb{E}_{p(\theta|\{\boldsymbol{x}_\mu\})}\left\{L(\theta, a)\right\} = \sum_{\theta \in \Theta} L(\theta, a) p(\theta|\left\{\boldsymbol{x}_\mu\right\}_\mu)$$

Bayesian decision theory (continued)

- The MAP is equivalent to minimizing a 0/1 loss

$$L(\hat{\theta}, \theta) = \mathbb{1}_{\theta \neq \hat{\theta}} = \begin{cases} 0 & \text{if } \hat{\theta} \neq \theta \\ 1 & \text{if } \hat{\theta} = \theta. \end{cases}$$

we then have

$$\begin{aligned} \mathbb{E}L(\hat{\theta}, \theta) = p(\theta \neq \hat{\theta} | \{\boldsymbol{x}_\mu\}_\mu) &= 1 - p(\hat{\theta} = \theta | \{\boldsymbol{x}_\mu\}_\mu) \\ &= 1 - p(\hat{\theta} = \theta | \{\boldsymbol{x}_\mu\}_\mu) p(\theta | \boldsymbol{x}_\mu) \end{aligned}$$

which is maximized for $\hat{\theta} = \theta$ with θ taken as

$$\theta^*(\{\boldsymbol{x}_\mu\}_\mu) = \underset{\hat{\theta}}{\text{argmax}}\ p(\theta | \{\boldsymbol{x}_\mu\}_\mu)$$

What do we do with noisy data?

- Is it possible to take more robust losses?
- ℓ_2 loss, $L(\hat{\theta}, \theta) = |\hat{\theta} - \theta|^2$ gives posterior mean

$$\mathbb{E}\left\{(\hat{\theta} - \theta)^2 | \boldsymbol{x}_\mu\right\} = \mathbb{E}[\theta^2 | \boldsymbol{x}_\mu] - 2\hat{\theta}\mathbb{E}[\theta | \boldsymbol{x}_\mu] + \hat{\theta}^2$$

- setting derivative to 0, $\partial_{\hat{\theta}}\mathbb{E}\{\hat{\theta} | \boldsymbol{x}_\mu\} = 0$, we get

$$-2\mathbb{E}\left\{\theta | \boldsymbol{x}_\mu\right\} + 2\hat{\theta} = 0$$

$$\hat{\theta} = \int \theta p(\theta | \boldsymbol{x}_\mu)\, d\theta$$

What do we do with noisy data? (continued)

- Is it possible to take more robust losses?
- ℓ_1 loss, $L(\hat{\theta}, \theta) = |\hat{\theta} - \theta|$ gives posterior median
- The value $\hat{\theta}$ such that

$$p(\theta < \hat{\theta} | \boldsymbol{x}_\mu) = p(\theta \geq \hat{\theta} | \boldsymbol{x}_\mu) = 1/2$$

What do we do with noisy data? (continued)

- Now assume $\hat{\theta}$ defines the value of some hidden variable y (e.g. the class of a point $\boldsymbol{x}_\mu$ defined by a gaussian mixture $\hat{\boldsymbol{\theta}}$).
- Finding the optimal parameters (or equivalently estimate the hidden state) can be done by considering the error

$$\begin{aligned} L_g(\theta, \hat{\theta}) &= \mathbb{E}_{(\boldsymbol{x}_\mu, y_\mu) \sim p(\boldsymbol{x}_\mu, y_\mu | \theta)} \left\{ \ell(\theta, f_{\hat{\theta}}) \right\} \\ &= \sum_{\boldsymbol{x}_\mu} \sum_{y_\mu} \ell(y_\mu, f_{\hat{\theta}}(x_\mu)) p(x_\mu, y_\mu | \boldsymbol{\theta}) \end{aligned}$$

- The Bayesian approach then minimizes the posterior expected loss

$$\underset{\hat{\theta}}{\operatorname{argmin}} \int p(\theta | \mathcal{D}) L_g(\theta, \hat{\theta}) \, d\theta$$

- Note that here the model is fixed and we want to learn the parameters ($><$ model selection)

How to pick up the priors?

- The controversial aspect of Bayesian statistics are the priors
- The main argument of Bayesians is that we most often know something about the world
- When it is possible, it makes things easier to pick up a prior from the same family as the likelihood function

Parameter/model inference: Biased vs Unbiased

- Imagine that we have access to a set of obervations $x_i \in \mathbb{R}^n$ and we can reasonably assume those samples are drawn independently from gaussian distributions.
- Because the observations are i.i.d, we can write the expression for the probability of oberving the x_i given the common μ and σ^2,

$$p(x|\mu, \sigma^2) = \prod_{n=1}^{N} \mathcal{N}(x_n|\mu, \sigma^2) \tag{2}$$

- A reasonably good idea to derive estimates for μ and σ is then to maximize this likelihood

Parameter/model inference: Biased vs Unbiased

- Since the log is a monotonically increasing function,

$$\underset{\mu,\sigma^2}{\text{argmax}} \quad p(x|\mu,\sigma^2) = \prod_{n=1}^{N} \mathcal{N}(x_n|\mu,\sigma^2)$$

is equivalent to maximizing the log likelihood function

$$\underset{\mu,\sigma^2}{\text{argmax}} \quad \log\left(p(x|\mu,\sigma^2)\right) = -\frac{1}{2\sigma^2}\sum_{n=1}^{N}(x_n-\mu)^2 - \frac{N}{2}\log(\sigma^2) - \frac{N}{2}\log(2\pi).$$

Parameter/model inference: Biased vs Unbiased

- Maximizing the log likelihood function with respect to μ first and then σ^2 gives the maximum likelihood estimators

$$\hat{\mu}_{ML} = \frac{1}{N}\sum_{n=1}^{N} x_n$$

$$\hat{\sigma}^2_{ML} = \frac{1}{N}\sum_{n=1}^{N}(x_n - \hat{\mu}_{ML})^2$$

- Those two estimates are functions of the data set $x_1, \ldots, x_N$

Parameter/model inference: Biased vs Unbiased

- Remember the ML estimators

$$\hat{\mu}_{ML} = \frac{1}{N}\sum_{n=1}^{N} x_n$$

$$\hat{\sigma}_{ML} = \frac{1}{N}\sum_{n=1}^{N}(x_n - \hat{\mu}_{ML})^2$$

- Now take the expectation of those estimators with respect to the known distribution,

$$\mathbb{E}\hat{\mu}_{ML} = \mu$$

$$\mathbb{E}\hat{\sigma}^2_{ML} = \left(\frac{N-1}{N}\right)\sigma^2$$

Parameter/model inference: Biased vs Unbiased

- On average, the MLE will get you the right mean, but will underestimate the variance

$$\mathbb{E}\hat{\mu}_{ML} = \mu$$
$$\mathbb{E}\hat{\sigma}^2_{ML} = \left(\frac{N-1}{N}\right)\sigma^2$$

- This problem is called bias and is related to the problem of overfitting
- In fact it turns out that a better estimator for σ^2 is given by

$$\hat{\sigma}^2 = \frac{1}{N-1}\sum_{n=1}^{N}(x_n - \hat{\mu}_{ML})^2$$