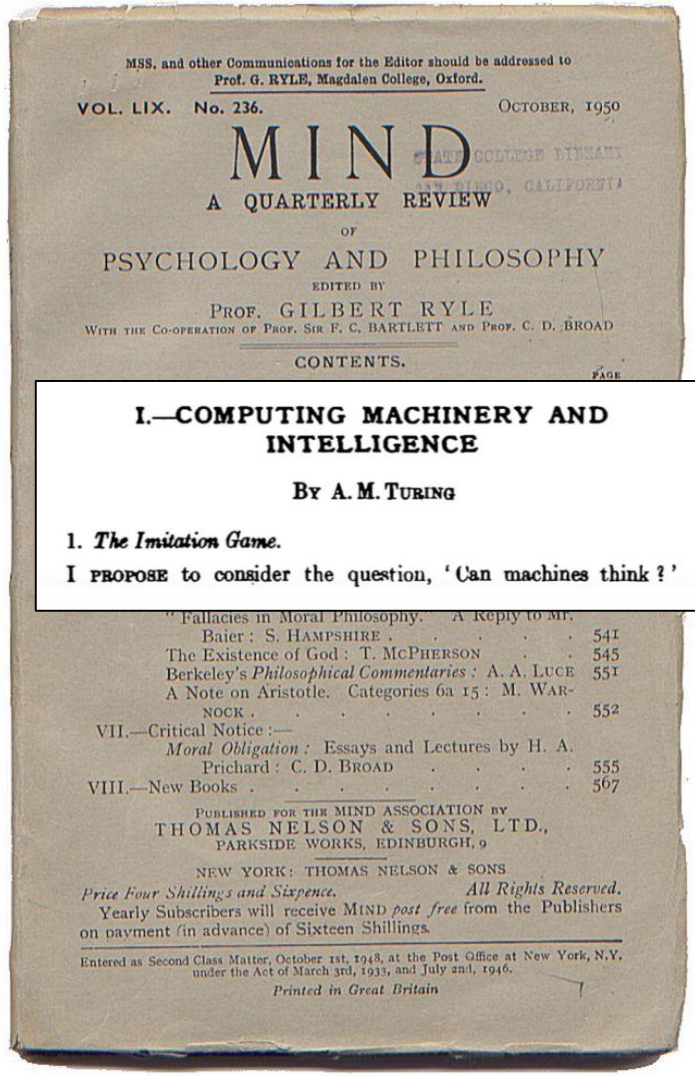
INTRODUCTION TO MACHINE LEARNING

14/09/2023

Origins of Machine Learning

ALAN TURING

1950



MSS. and other Communications for the Editor should be addressed to
Prof. G. RYLE, Magdalen College, Oxford.

VOL. LIX. No. 236. OCTOBER, 1950

MIND

A QUARTERLY REVIEW

OF

PSYCHOLOGY AND PHILOSOPHY

EDITED BY

PROF. GILBERT RYLE

WITH THE CO-OPERATION OF PROF. SIR F. C. BARTLETT AND PROF. C. D. BROAD

CONTENTS.

PAGE

I.—COMPUTING MACHINERY AND INTELLIGENCE

BY A. M. TURING

1. *The Imitation Game.*

I PROPOSE to consider the question, 'Can machines think?'

"Fallacies in Moral Philosophy." A Reply to Mr. Baier: S. HAMPSHIRE 541

The Existence of God: T. MCPHERSON . . . 545

Berkeley's *Philosophical Commentaries*: A. A. LUCE 551

A Note on Aristotle. Categories 6a 15: M. WARNOCK 552

VII.—Critical Notice:—

Moral Obligation: Essays and Lectures by H. A. Prichard: C. D. BROAD 555

VIII.—New Books 567

PUBLISHED FOR THE MIND ASSOCIATION BY
THOMAS NELSON & SONS, LTD.,
PARKSIDE WORKS, EDINBURGH, 9

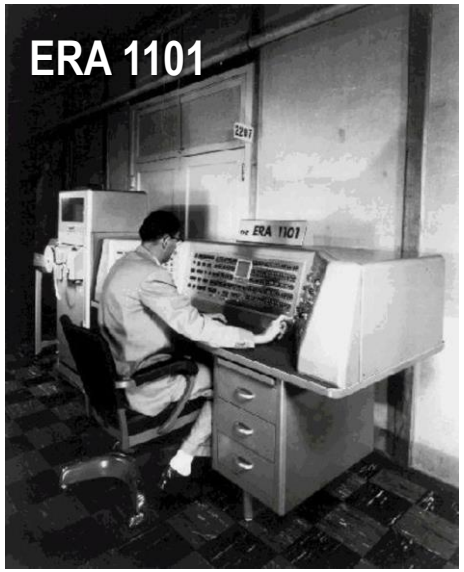
NEW YORK: THOMAS NELSON & SONS

Price Four Shillings and Sixpence. *All Rights Reserved.*

Yearly Subscribers will receive MIND *post free* from the Publishers on payment (in advance) of Sixteen Shillings.

Entered as Second Class Matter, October 1st, 1948, at the Post Office at New York, N.Y. under the Act of March 3rd, 1933, and July 2nd, 1946.

Printed in Great Britain

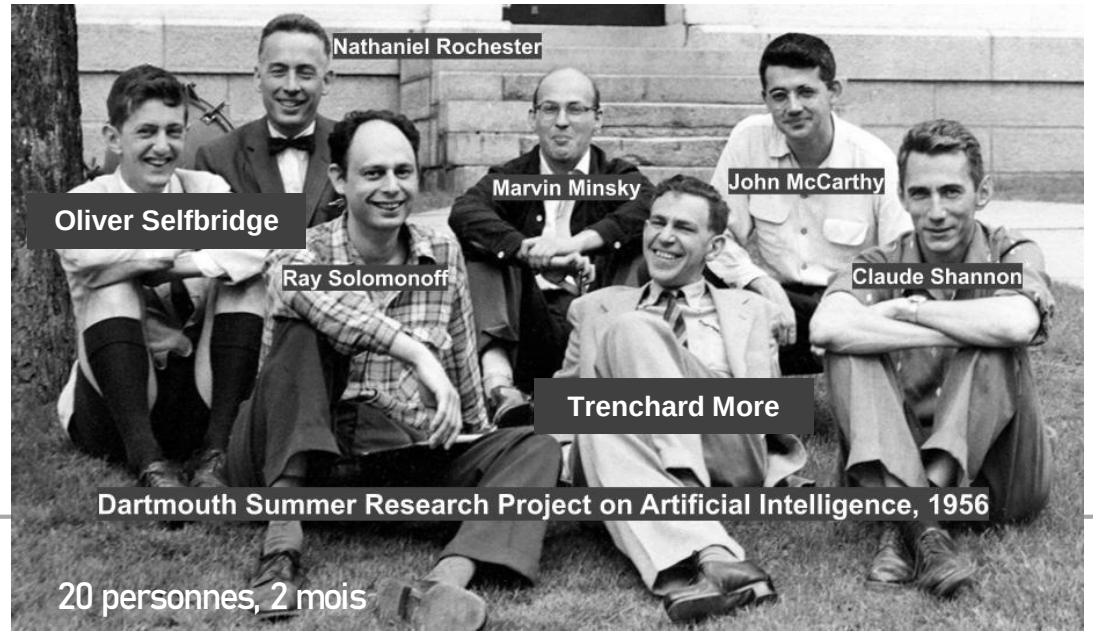


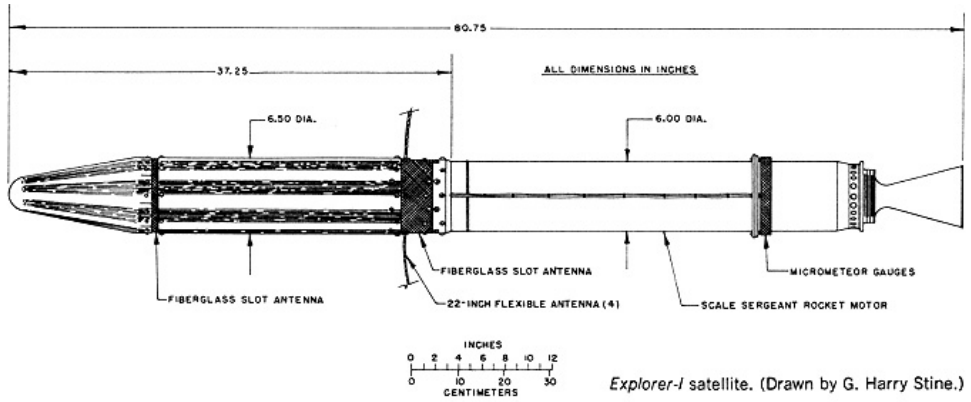
Origins of Machine Learning

DARMOUTH CONFERENCE

1950

1956





Explorer 1, James Van Allen's Geiger counter

Artificial Intelligence, Machine Learning and Deep Learning

- If the term "artificial intelligence" (AI) has become commonplace in the media, there is no real shared definition of it

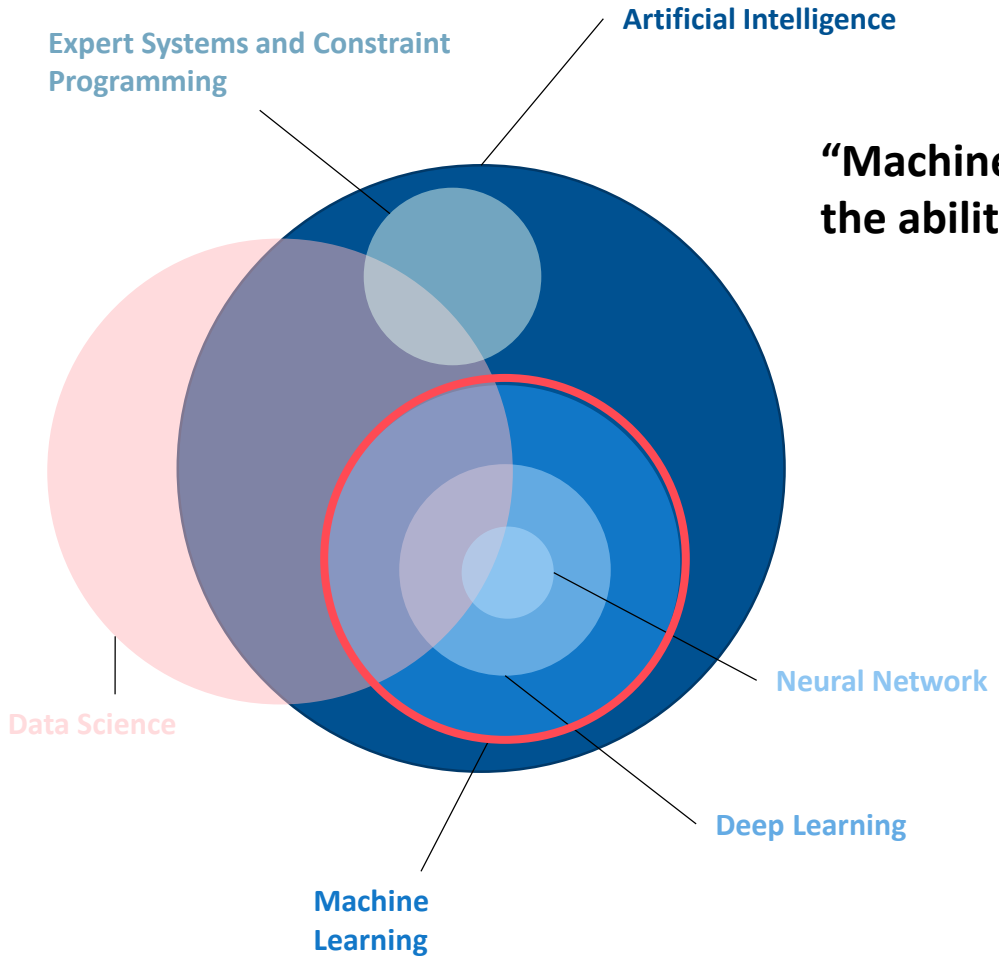
> *"Artificial intelligence (AI) refers to systems designed by humans that, given a complex goal, act in the physical or digital world by perceiving their environment, interpreting the collected structured or unstructured data, reasoning on the knowledge derived from this data and deciding the best action(s) to take (according to pre-defined parameters) to achieve the given goal.*
>
> *AI systems can also be designed to learn to adapt their behaviour by analysing how the environment is affected by their previous actions."*

The High-Level Independent Expert Group on Artificial Intelligence of the European Commission defined artificial intelligence systems on April 8, 2019

We can distinguish the following in Artificial Intelligence :

- Machine learning (including deep learning and reinforcement learning as specific examples)
- Automated reasoning (including planning, programming, knowledge representation and reasoning, search, and optimization)
- Robotics (including control, perception, sensors and actuators, as well as the integration of all other techniques into cyber-physical systems)

Artificial Intelligence, Machine Learning and Deep Learning



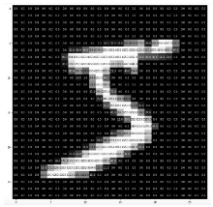
“Machine learning is the field of study that gives computers the ability to learn without being explicitly programmed”

Arthur L. Samuel, AI pioneer, 1959

What is Machine Learning ?

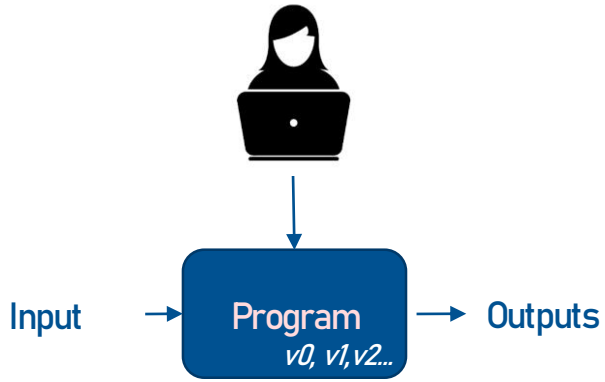
« Traditional » Approach

Hypothetico-deductive machine



Explicit code to give « 5 »

→ 5



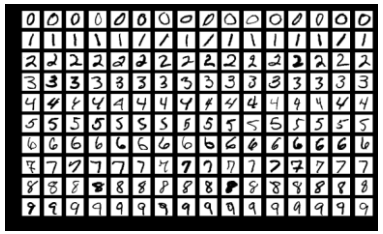
“Machine learning is the field of study that gives computers the ability to learn without being explicitly programmed”

Arthur L. Samuel, AI pioneer, 1959

Symbolism

« Machine Learning » Approach

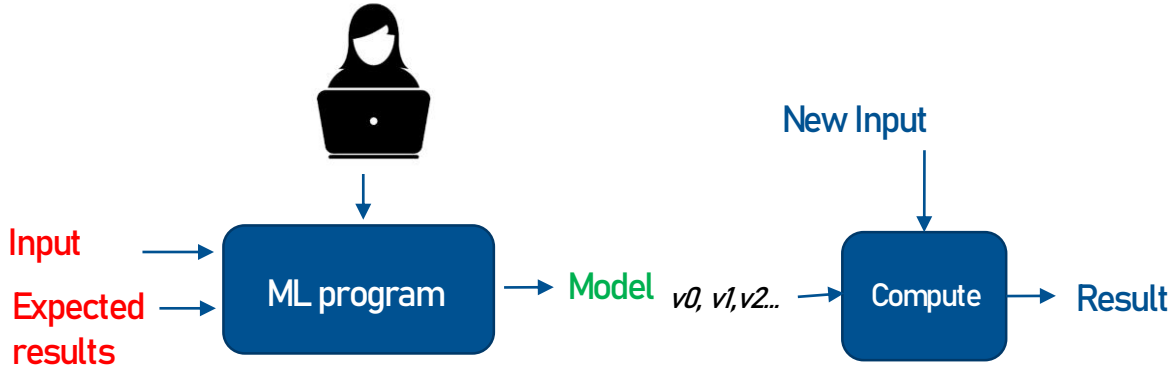
Inductive machine



Code to create a model

→ Model

Connexionnism



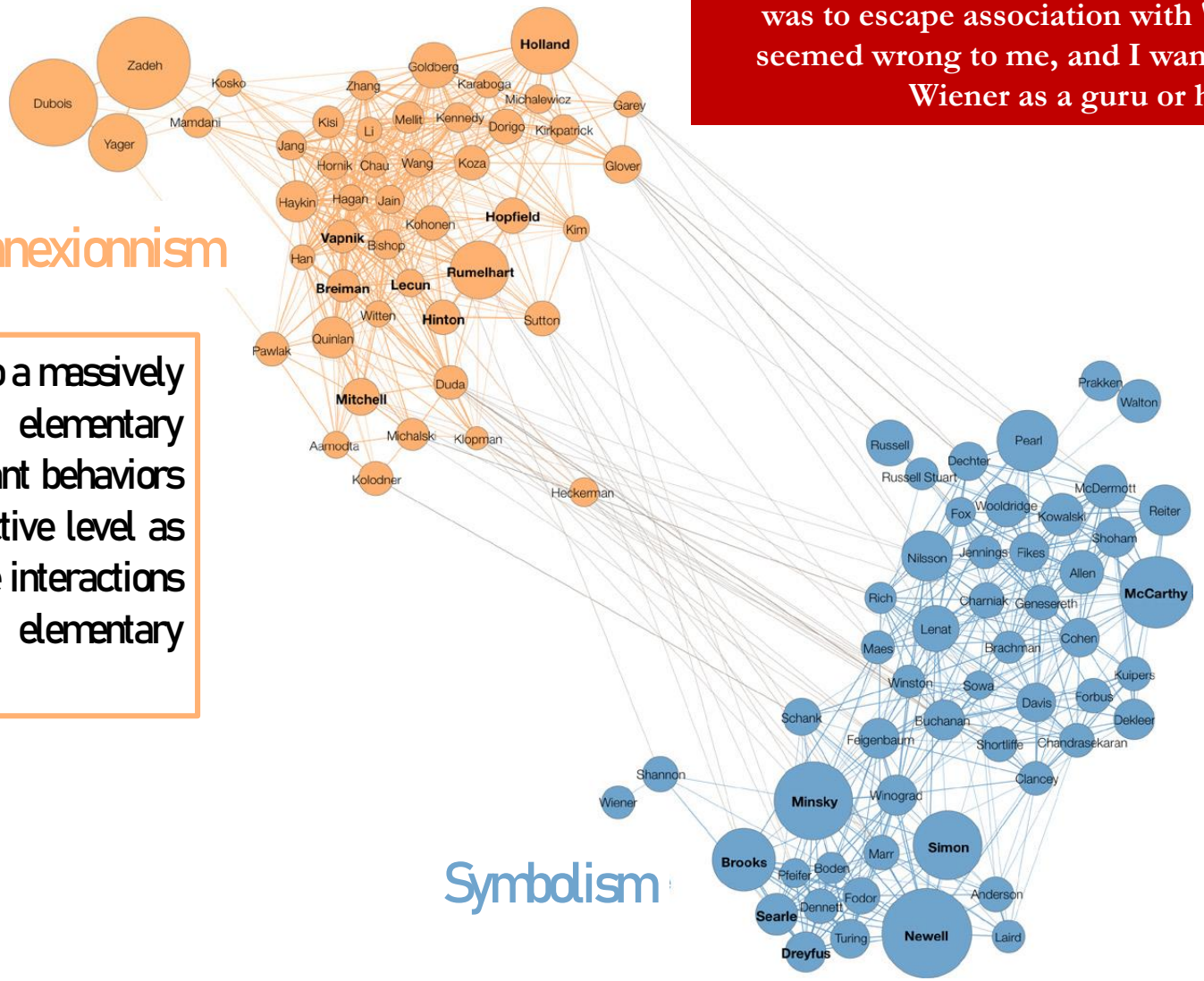
Connexionnism and Symbolism

« As for me, one of the reasons I invented the term 'artificial intelligence' was to escape association with 'cybernetics.' This focus on feedback seemed wrong to me, and I wanted to avoid having to accept Norbert Wiener as a guru or having to argue with him. »

John McCarthy (1988)

Connexionnism

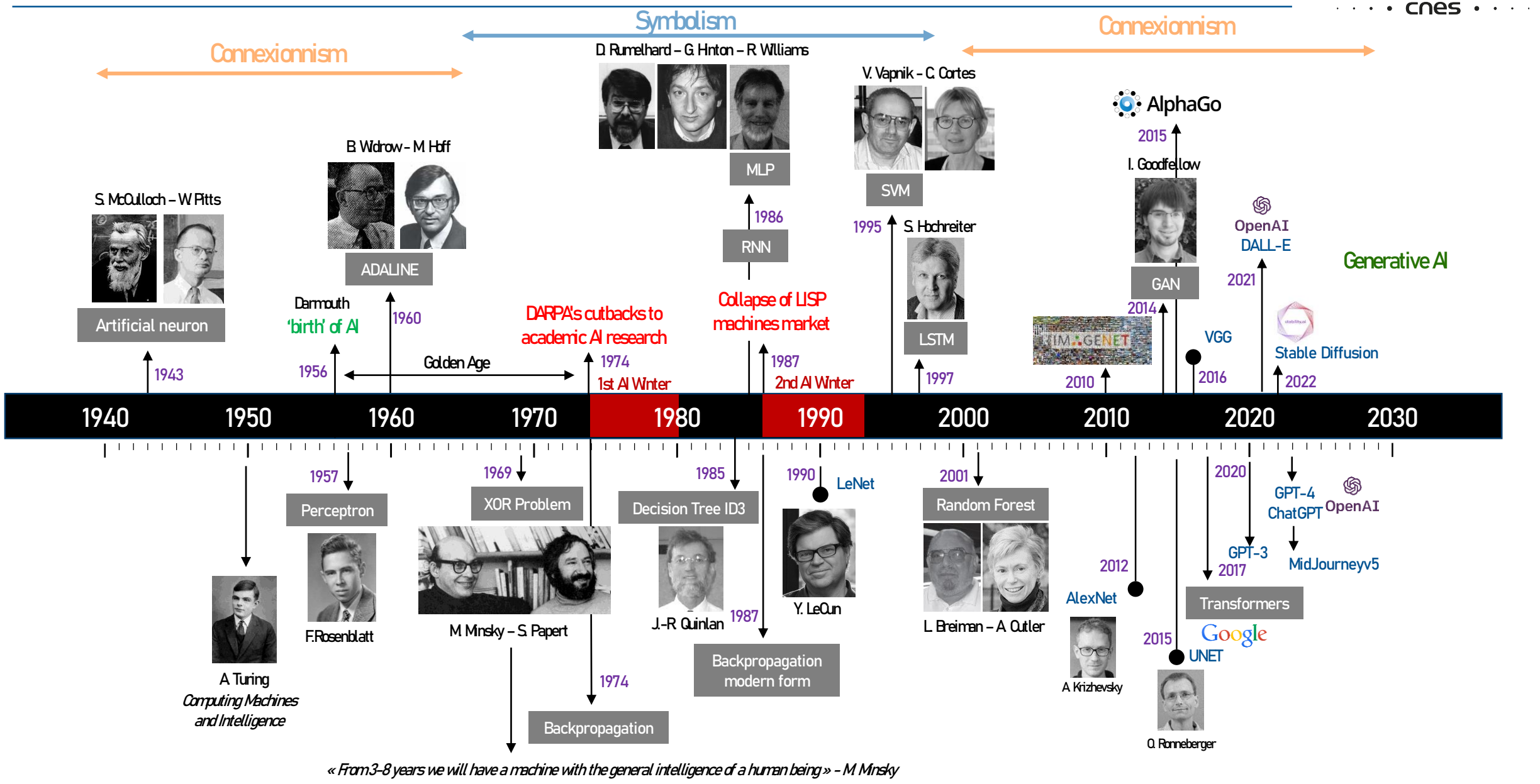
Thinking can be likened to a massively parallel calculation of elementary functions, whose significant behaviors only appear at the collective level as an emergent effect of the interactions produced by these elementary operations.



Symbolism

According to the orthodox cognitivism, thinking is equivalent to calculating symbols that have both a material reality and a semantic value of representation

Network of co-citations of the 100 most cited authors in scientific publications mentioning 'Artificial Intelligence'



« From 3-8 years we will have a machine with the general intelligence of a human being » - M. Minsky

Why doing Machine Learning ?

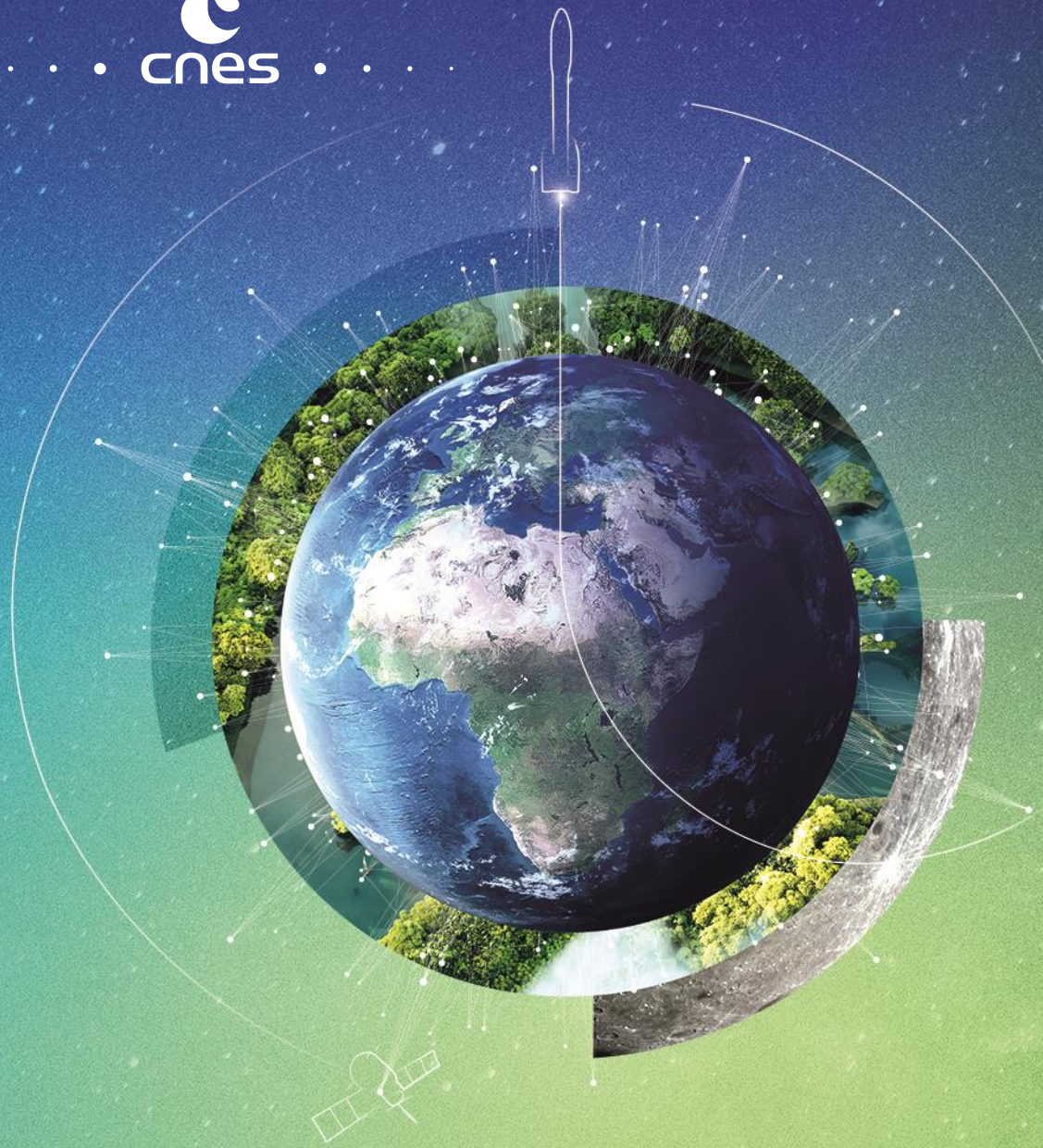
- Machine learning can be used to solve problems:
  - that we do not know how to solve
  - that we know how to solve, but do not know how to formalize in algorithmic terms how we solve them (as is the case, for example, with image recognition or natural language understanding)
  - that we know how to solve, but with procedures that are far too resource-intensive (as is the case, for example, with the prediction of interactions between large molecules, for which simulations are very heavy)

Machine learning is therefore used when data is abundant (relatively), but knowledge is not easily accessible or developed

Machine learning can also help humans learn: models created by learning algorithms can reveal the relative importance of certain information or how it interacts to solve a particular problem

Machine learning is used when it is difficult or impossible to define explicit instructions to give to a computer to solve a problem, but when there are many illustrative examples available

cnes



AI for

MONITORING

Earth

AI for

WATCHING

space

AI for

AUTONOMY

of the systems

AI for

BOARDABILITY

of artificial intelligence

AI for

ADAPTING

to a new world

AI for

MONITORING

Earth

Urban

Agriculture

Ecology

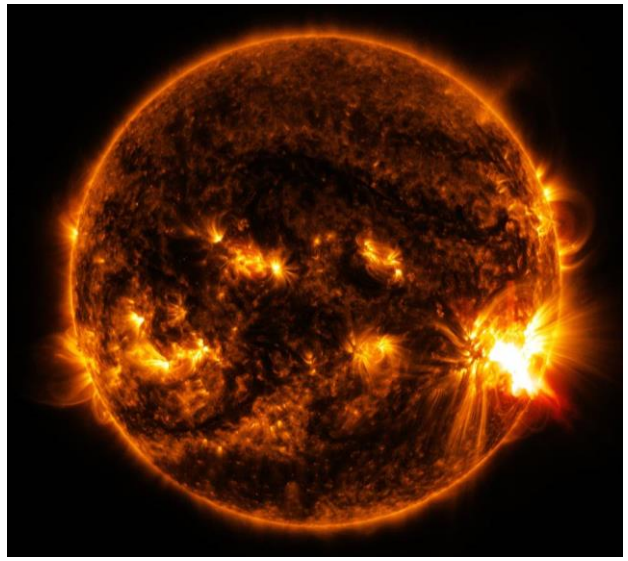
Climate Change

Image Quality

AI for

WATCHING

space



© NASA Goddard Space Flight Center

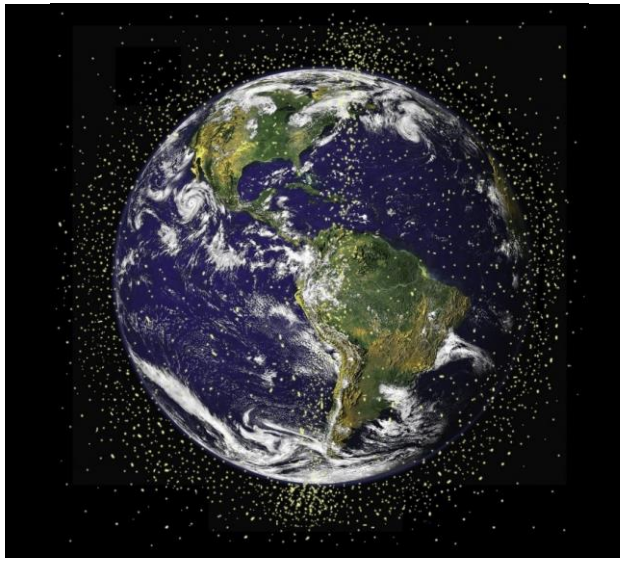
Space Meteorology

Sun monitoring and impacts

© ESA

Natural Objects

Detect potential earth-impacting objects



© NASA

Space Monitoring

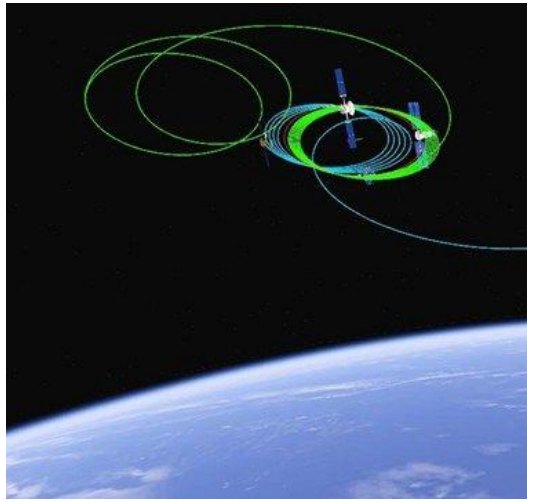
Keeping tracks of satellites and debris

© NASA Goddard Space Flight Center

IA for

AUTONOMY

of the systems



GNC

Guidage, Contrôle et Navigation

Operation Center

of the future

Exploration

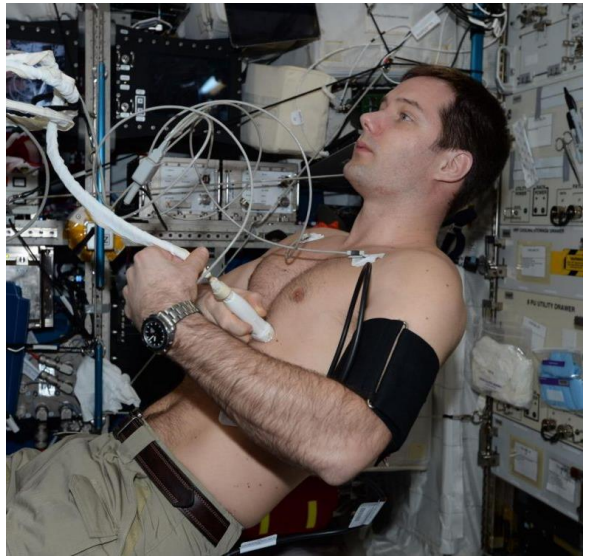
Bases, rendez-vous

AI for

ASSISTING

astronauts

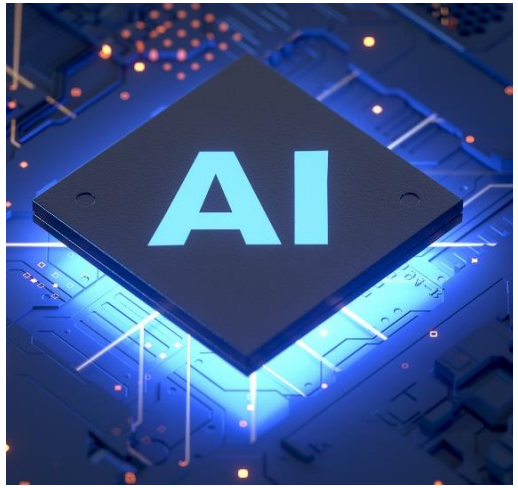
Virtual Assistant
Assist astronauts



Medical
Medical autonomy

BOARDING

Artificial Intelligence

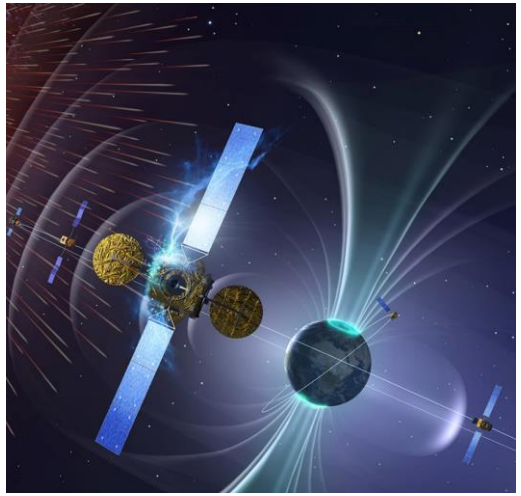


© INTEL

Prepare
the future

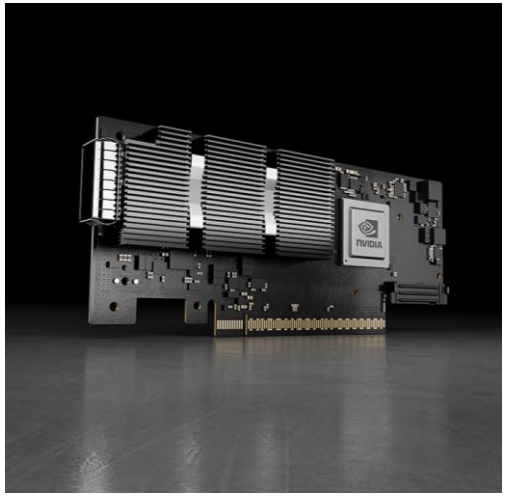
© ESA

Augment
boarding autonomy



© ESA

Adapt
to constraint

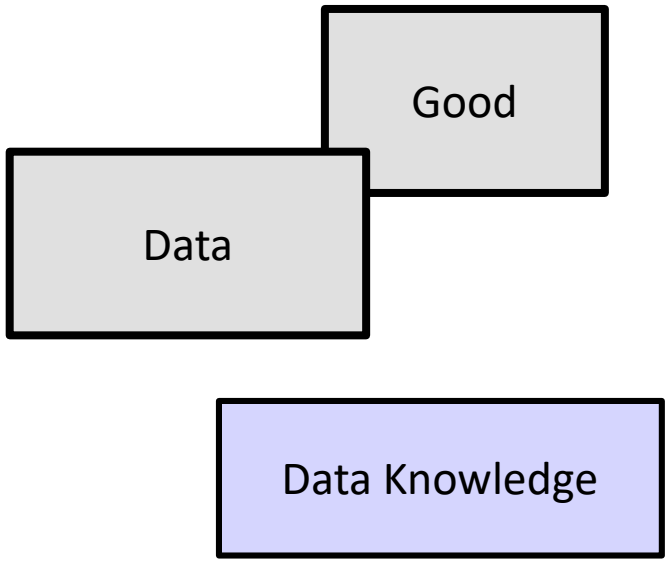


© NVIDIA

Scaling
AI

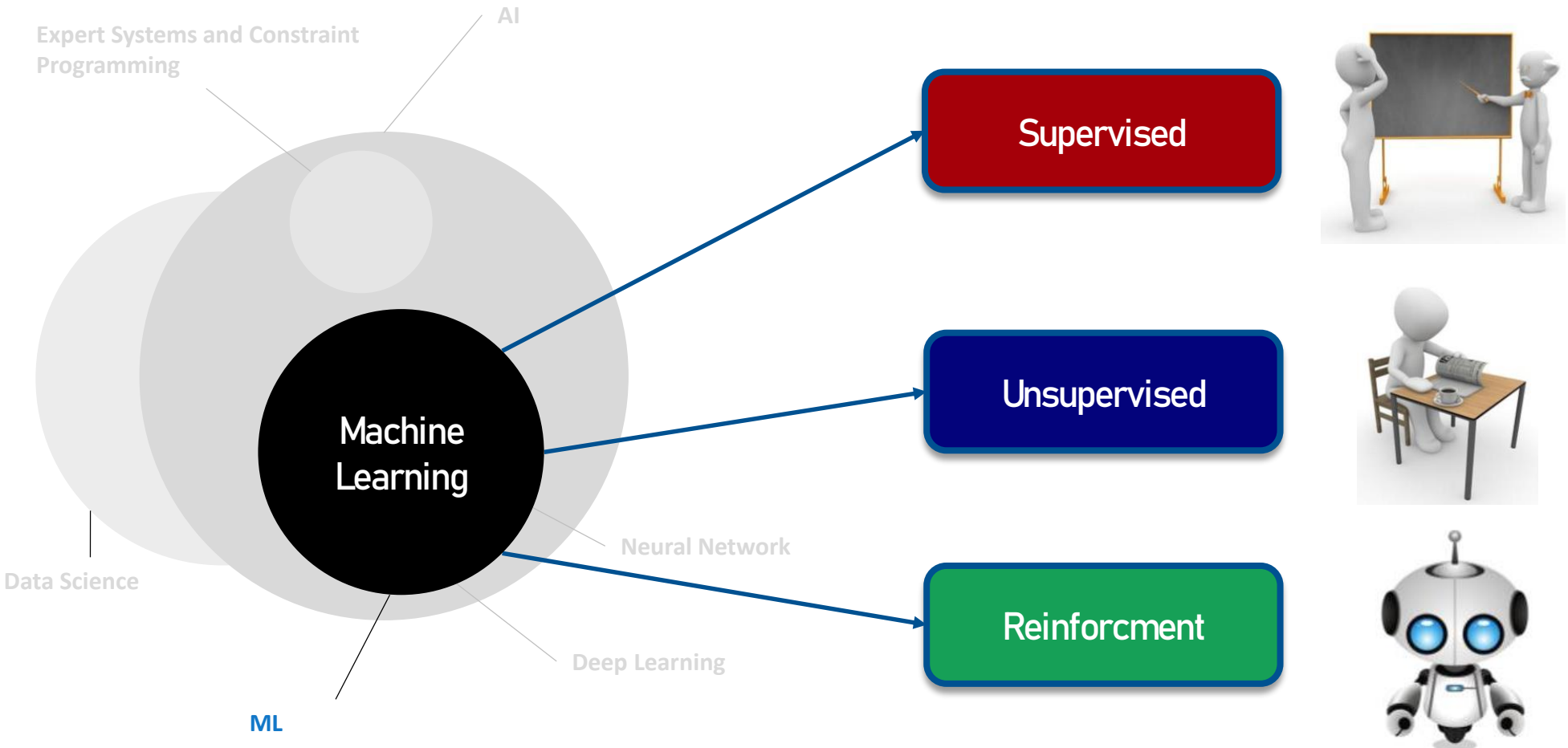
What an algorithm of Machine Learning needs

To find the right hypothesis and consolidate it, the following steps are necessary

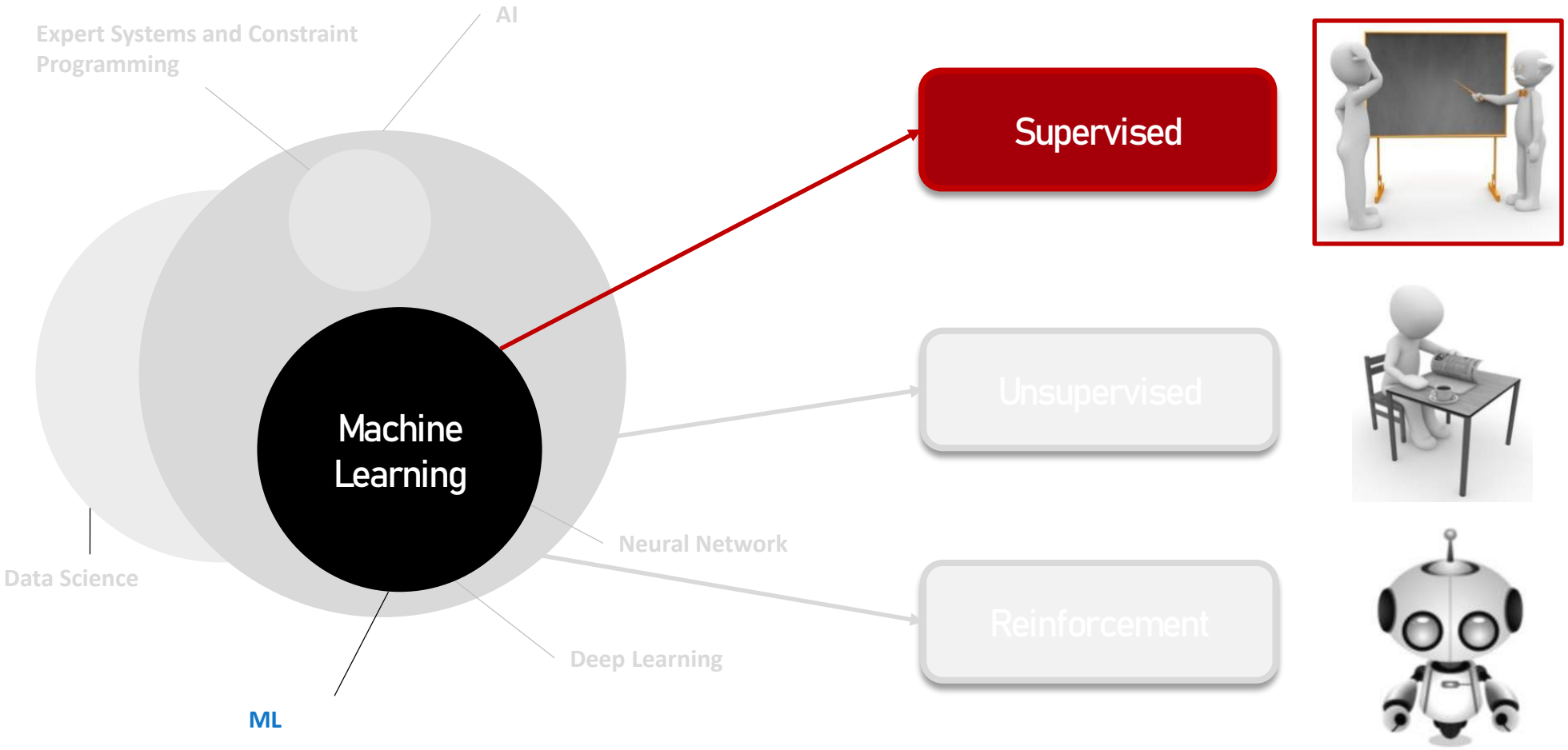


- There seems to be a repetitive pattern in the data.
- The problem can be solved analytically, but at a significant cost (or not)
- There is a sufficient corpus of information (and cleaned)
- The information is labeled by a human (or not)

Different ML models

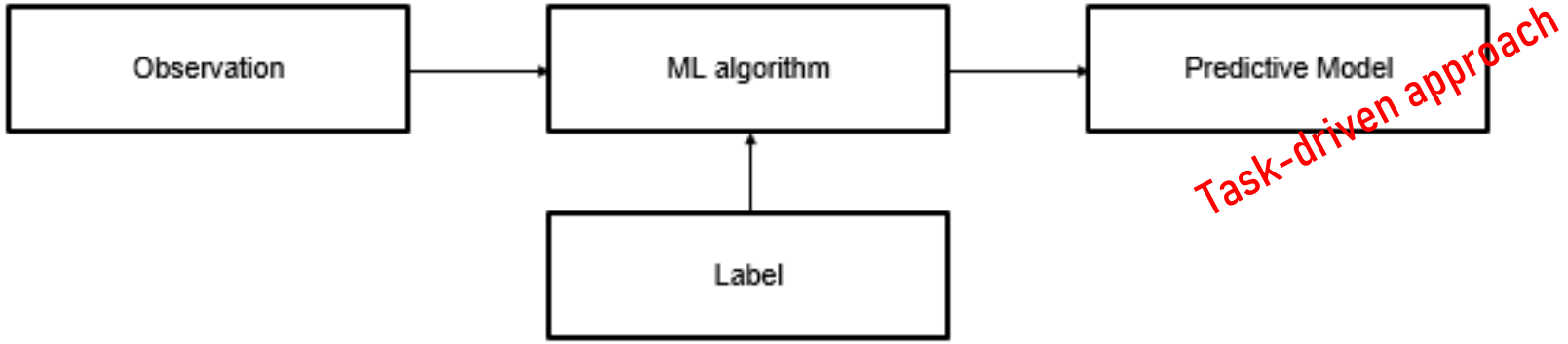


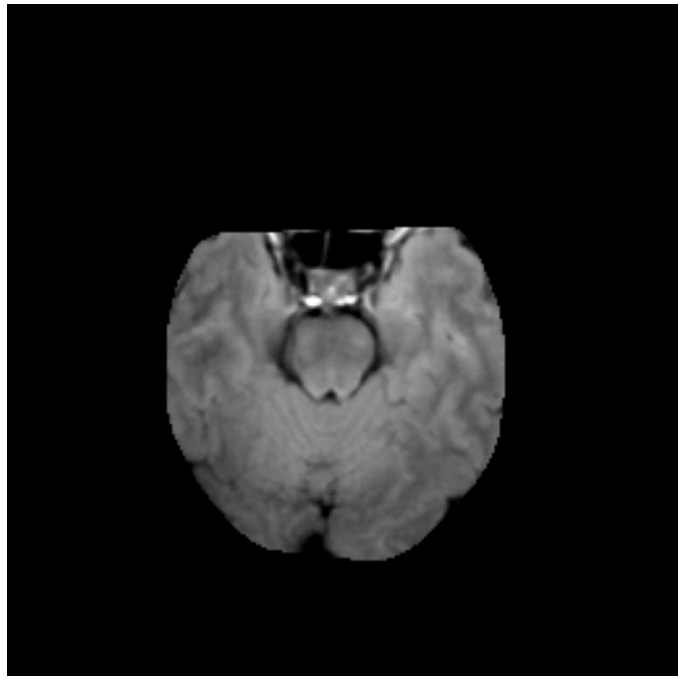
Supervised Machine Learning



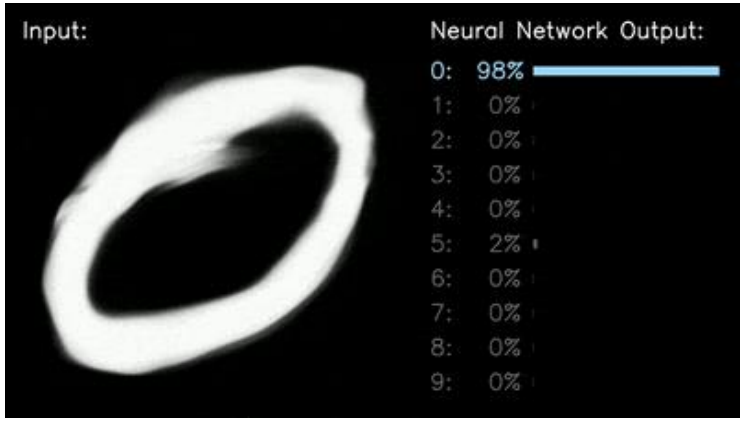
Supervised Machine Learning

- We give the algorithm a certain number of examples (inputs) to learn from, and these examples are 'labeled', meaning that they are associated with a desired result (output).
- The algorithm's task is then to find the law that allows it to find the output based on the inputs.

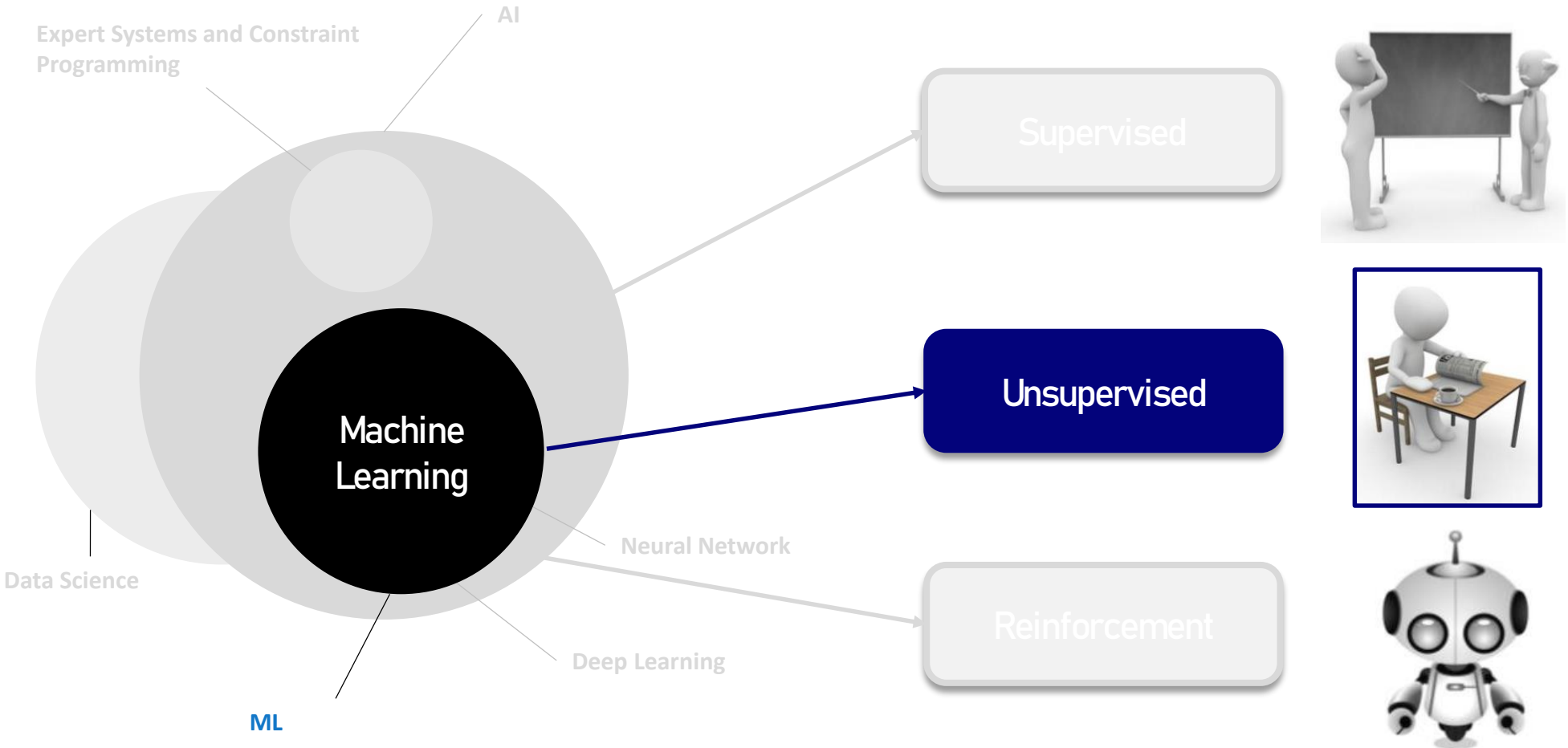




- A typical example is classification/segmentation:
  - Medical diagnosis can be established: is a patient suffering from a tumor or not (output), based on their symptoms and phenotypes (inputs).
  - Another example: recognition of handwritten digits (is the image a "1" or a "9"?). In this case, the input is a set of pixels, and the output takes 10 values, from 0 to 9.

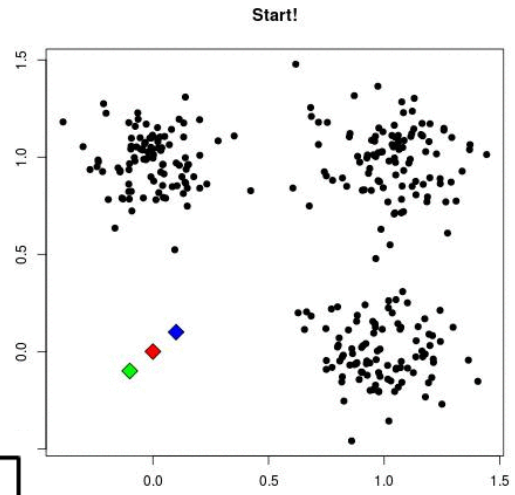


Unsupervised

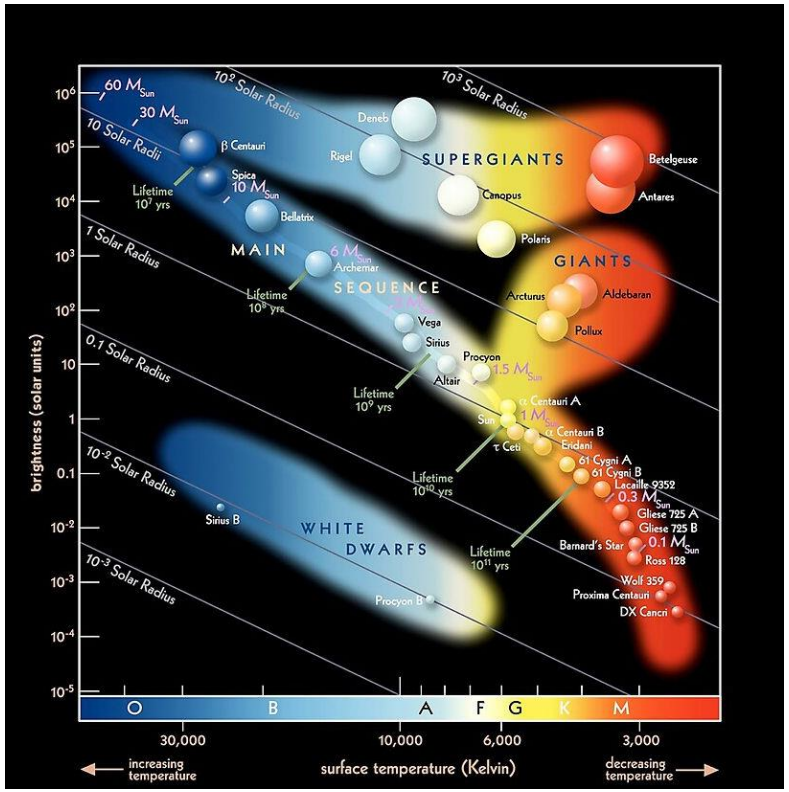


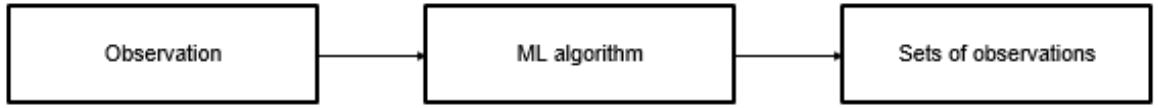
Unsupervised

Data-driven approach

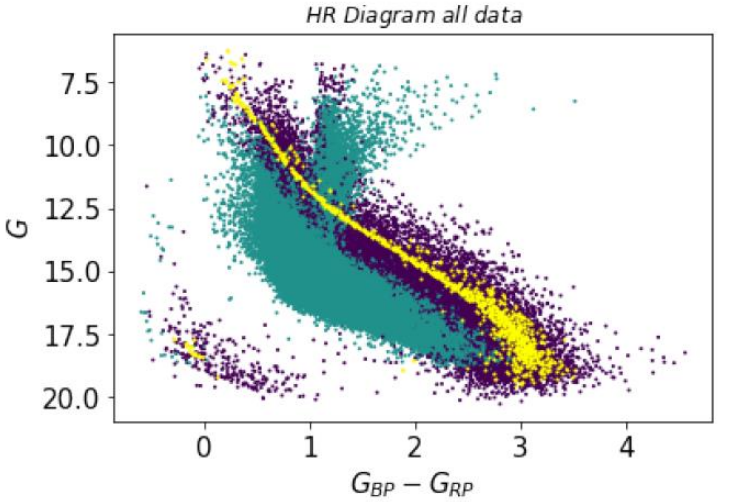


- Here, no label is provided to the algorithm and it must discover without human assistance the characteristic structure of the input.
- The typical example is clustering: the algorithm will group the examples into different clusters or classes.

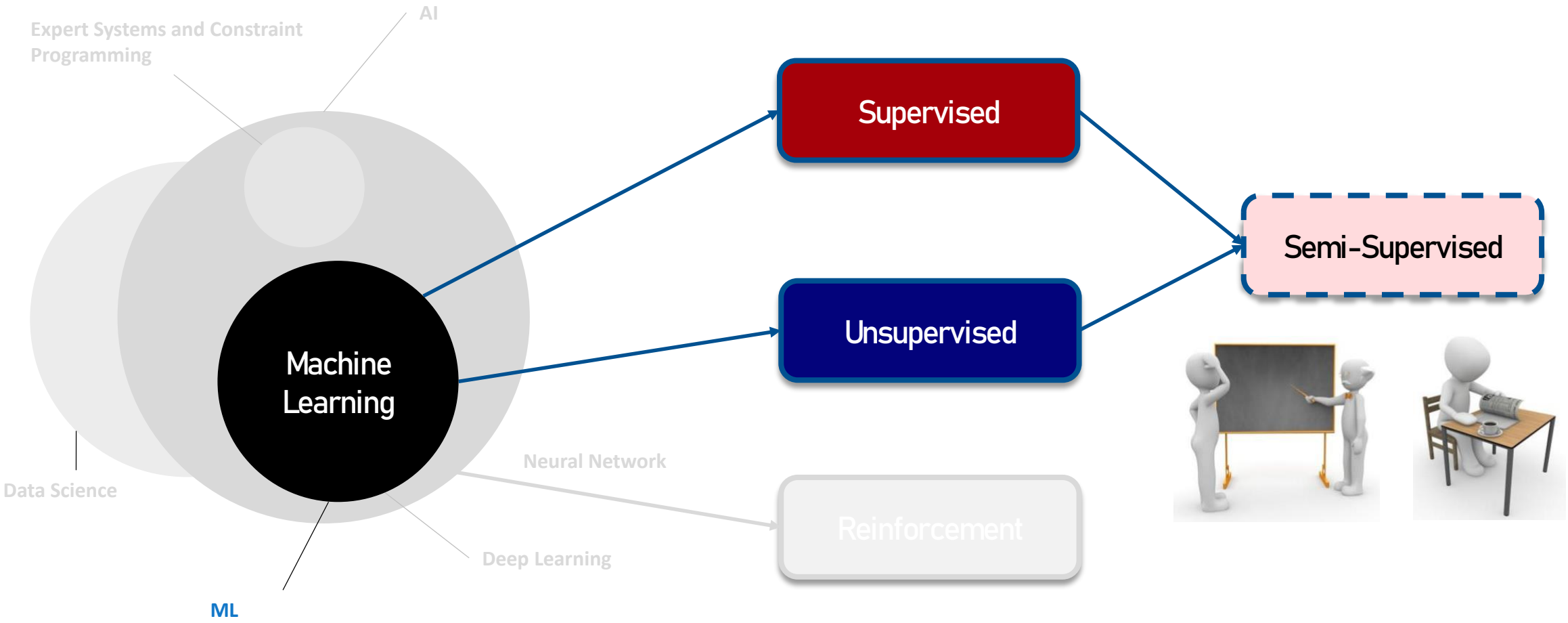




- Example : « *clusterisation* » of a Hertzsprung-Russell diagram



Les différents modèles de ML

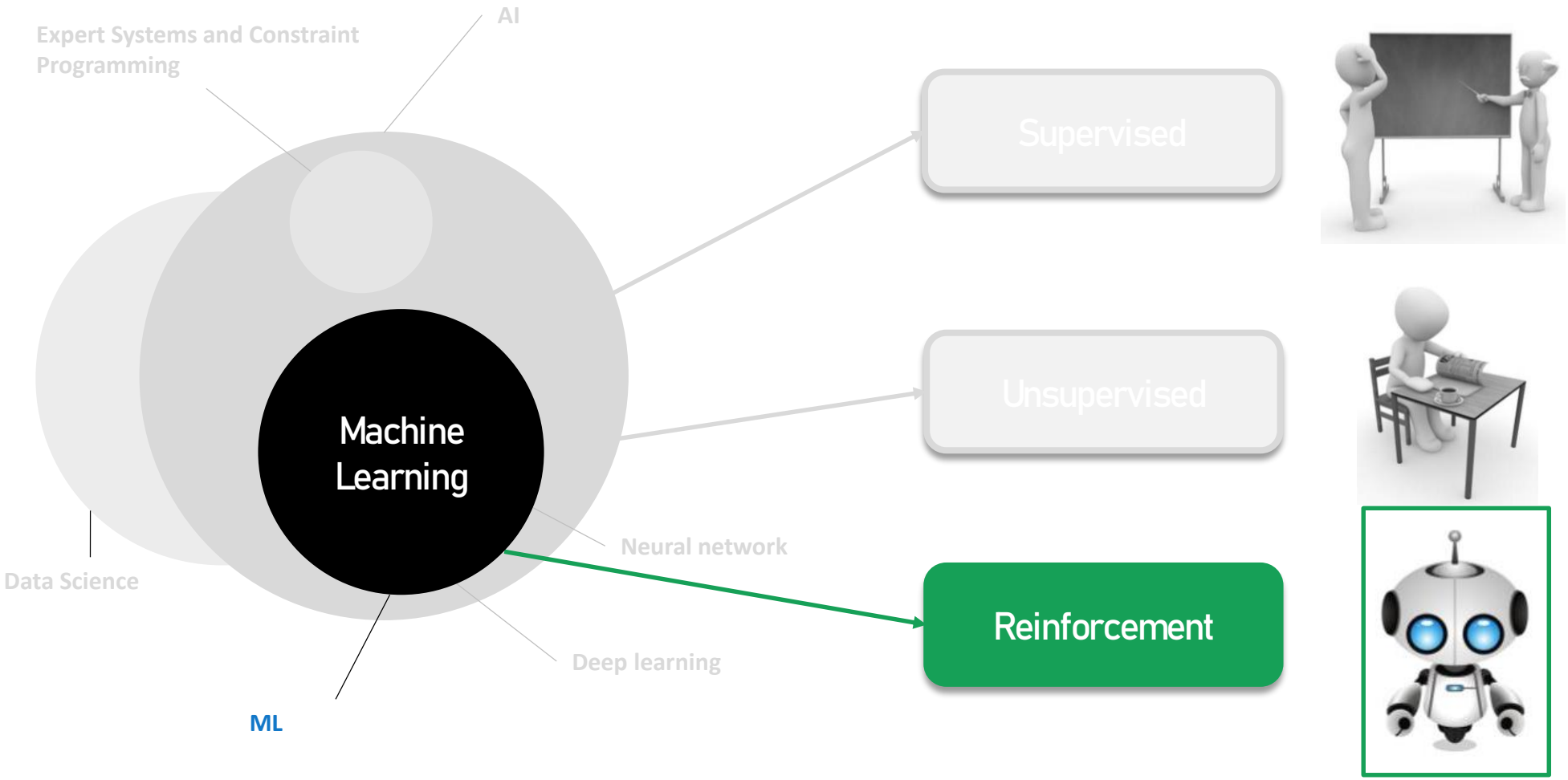


Semi-Supervised ML

- Semi-supervised learning can be defined as a hybridization of supervised and unsupervised methods.
- It uses both :
  - Labeled data
  - Unlabeled data
- Thus, it is situated between "unsupervised" and "supervised" learning.
- In the real world, labeled data can be quite rare in certain contexts, while unlabeled data is abundant, hence the interest in semi-supervised learning
- The ultimate goal of a semi-supervised learning model is to provide a better prediction result than that produced using only the labeled data of the model
- Some application areas where semi-supervised learning is used include:
  - Automatic translation,
  - Fraud detection,
  - Data labeling,
  - Text classification...

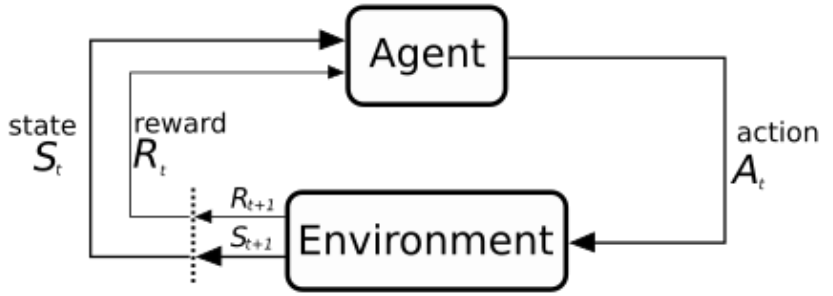
Hybridation

Reinforcment Learning

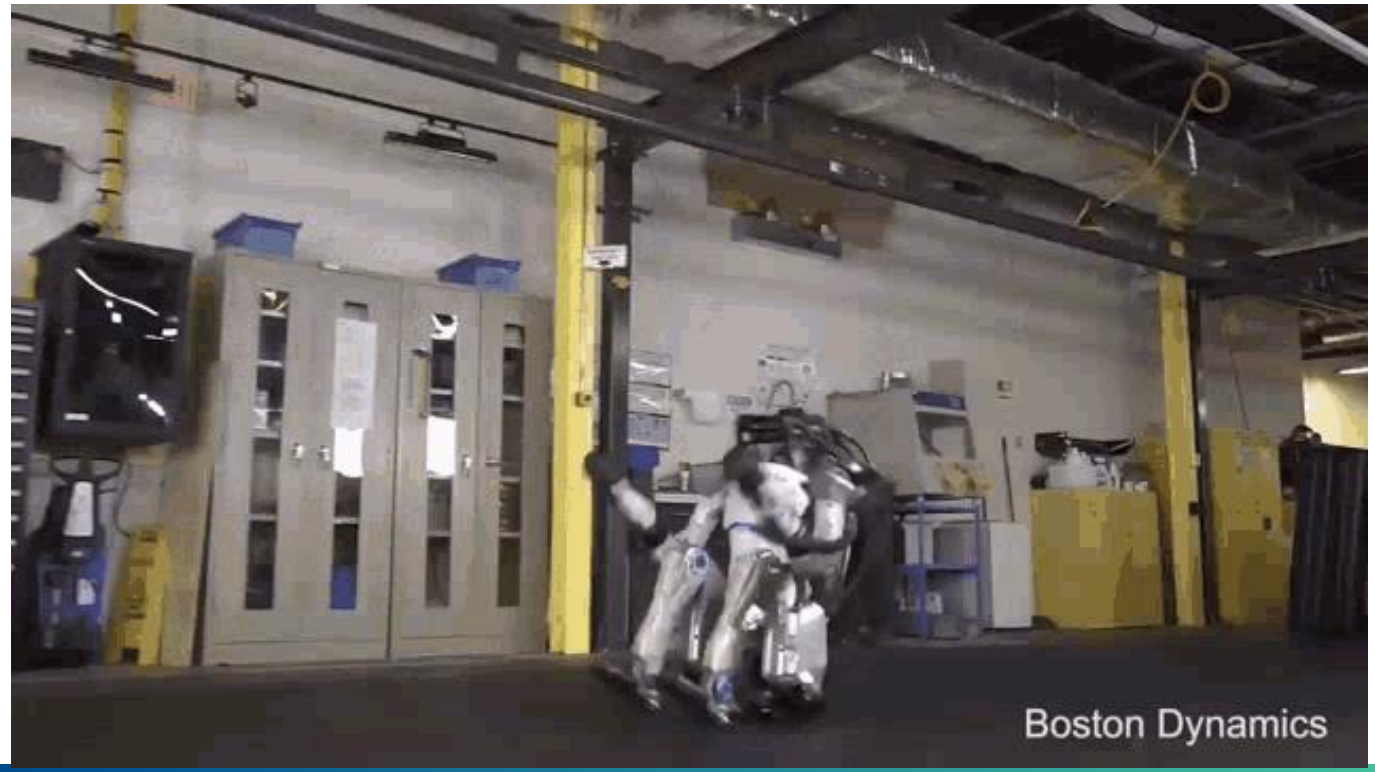


Reinforcment Learning

Environment-driven approach



- Reinforcment is a kind of semi-supervised learning
- The algorithm is designed to seek to optimize a quantitative reward, positive or negative, at all costs, based on experiences corresponding to different situations
- One of the most recent and impressive examples of reinforcement learning is the robot from Boston Dynamics:
  - Boston Dynamics has developed a robot capable of walking, running, climbing, and carrying heavy loads
  - The robot is trained to walk through reinforcement: its reward is strong and positive if it stays upright, its reward is negative if it falls
  - It is programmed to explore the different movements it can perform and conduct its own experiments
  - It is thus that it learns on its own to climb a slope and optimize the way it places its legs (speed, frequency, angle, etc.) in a way very similar to a human



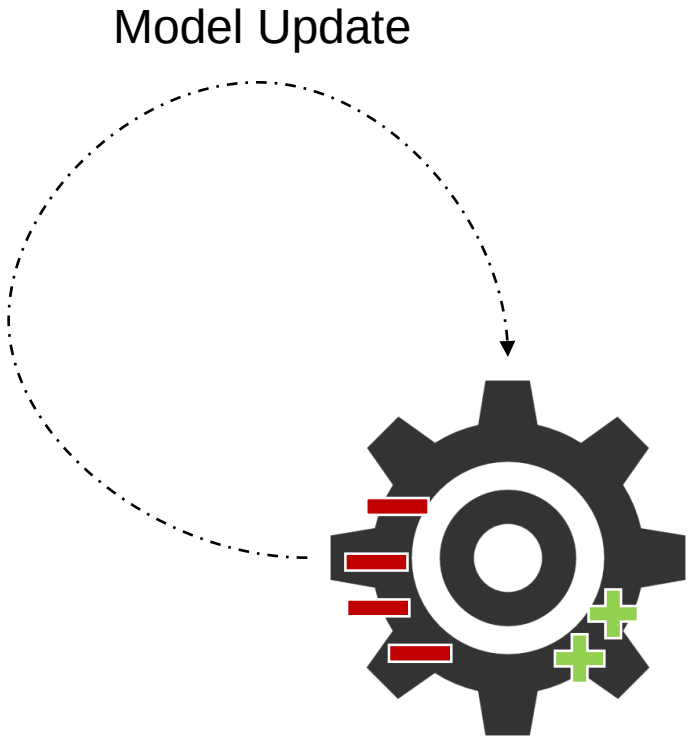
Supervised Machine Learning

Principle of Machine Learning

$$\hat{f}(\quad) = \quad cat$$

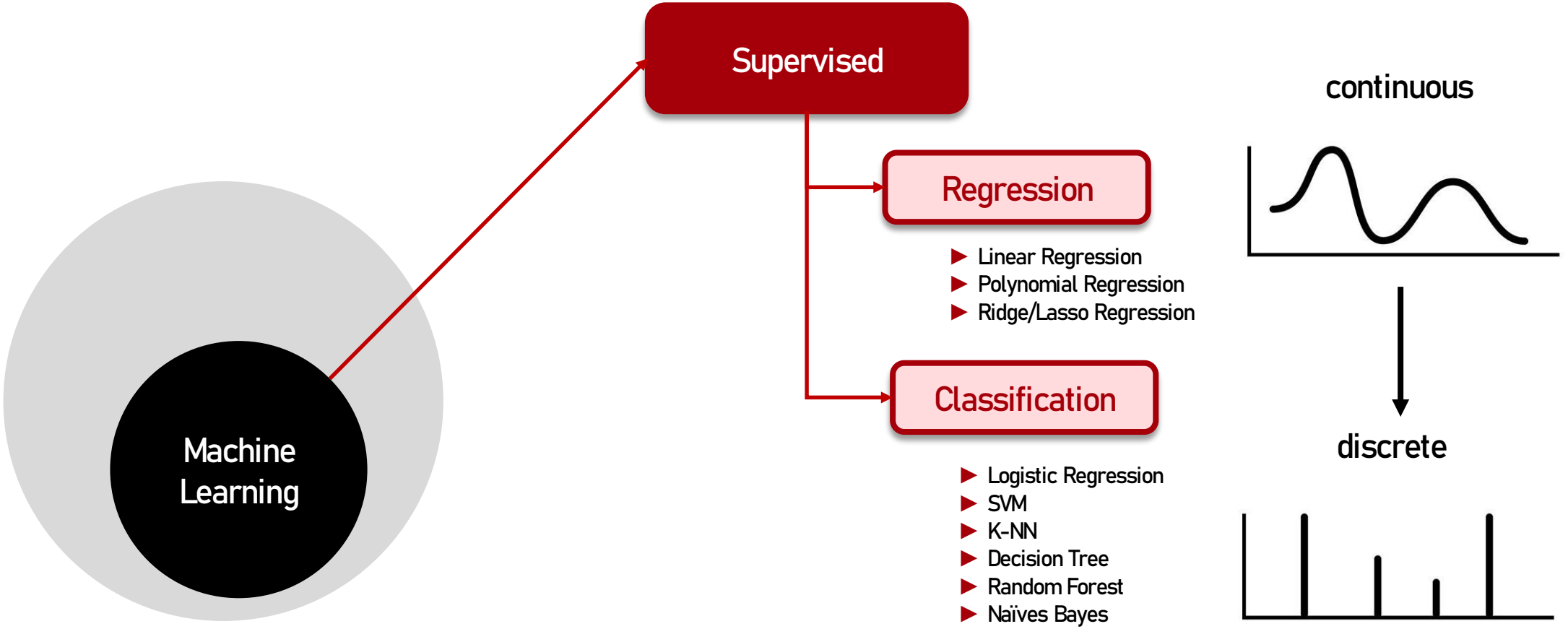
Objective: Find a function $\hat{f}$ approximating f

Principle of Machine Learning

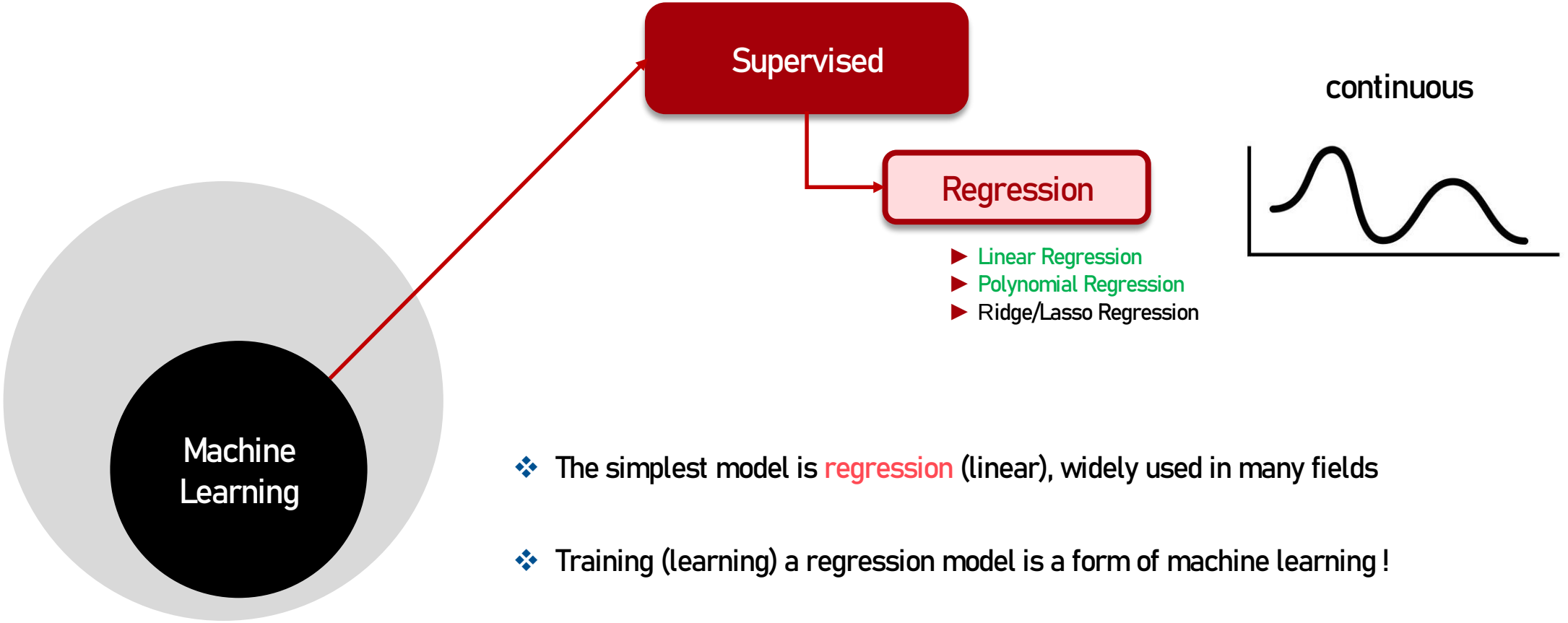


$$\hat{f}(x) = cat$$

Supervised



Regression



- The simplest model is regression (linear), widely used in many fields
- Training (learning) a regression model is a form of machine learning !
- We can trace back machine learning to the 17th century with Legendre and Gauss and their method of least squares

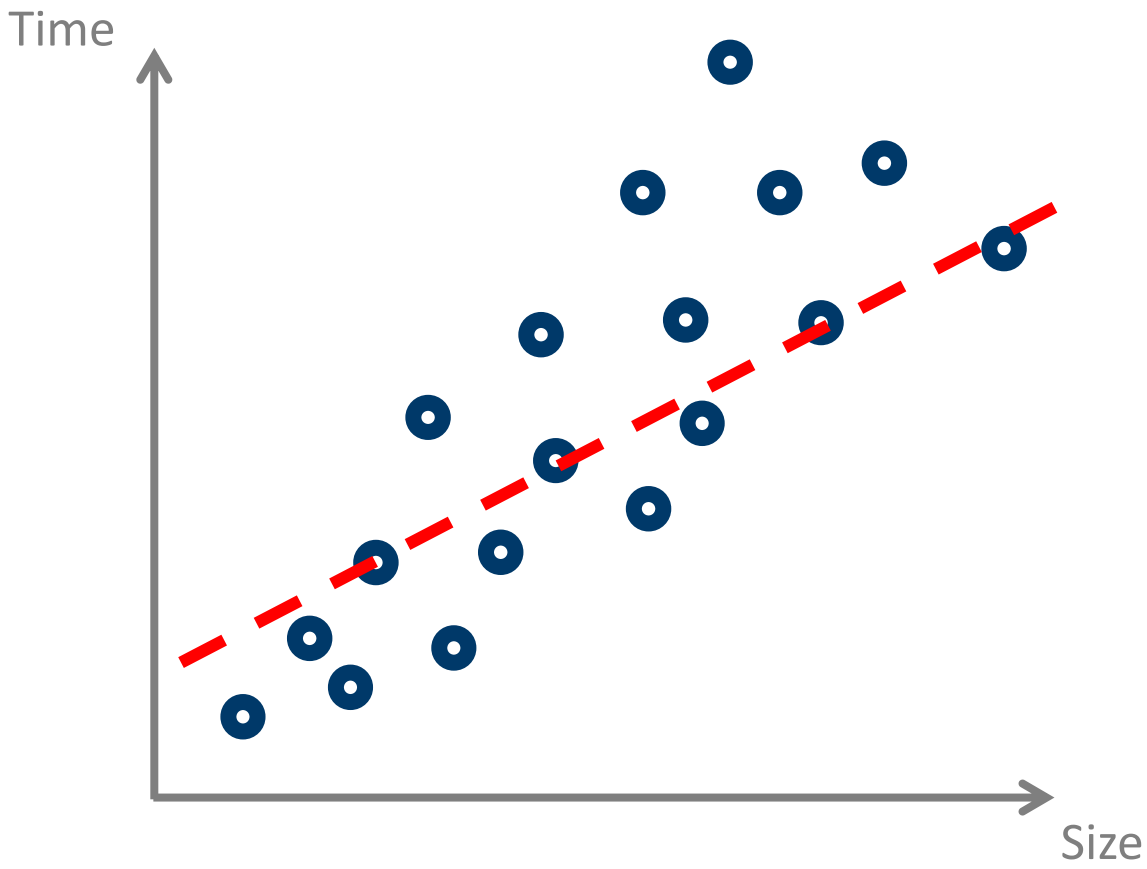
Regression

How much time to download a file ?

Regression

Size	Time to download
x	y
0,2 Go	2 min
1,0 Go	60 min
4,5 Go	45 min
10,0 Go	120 min

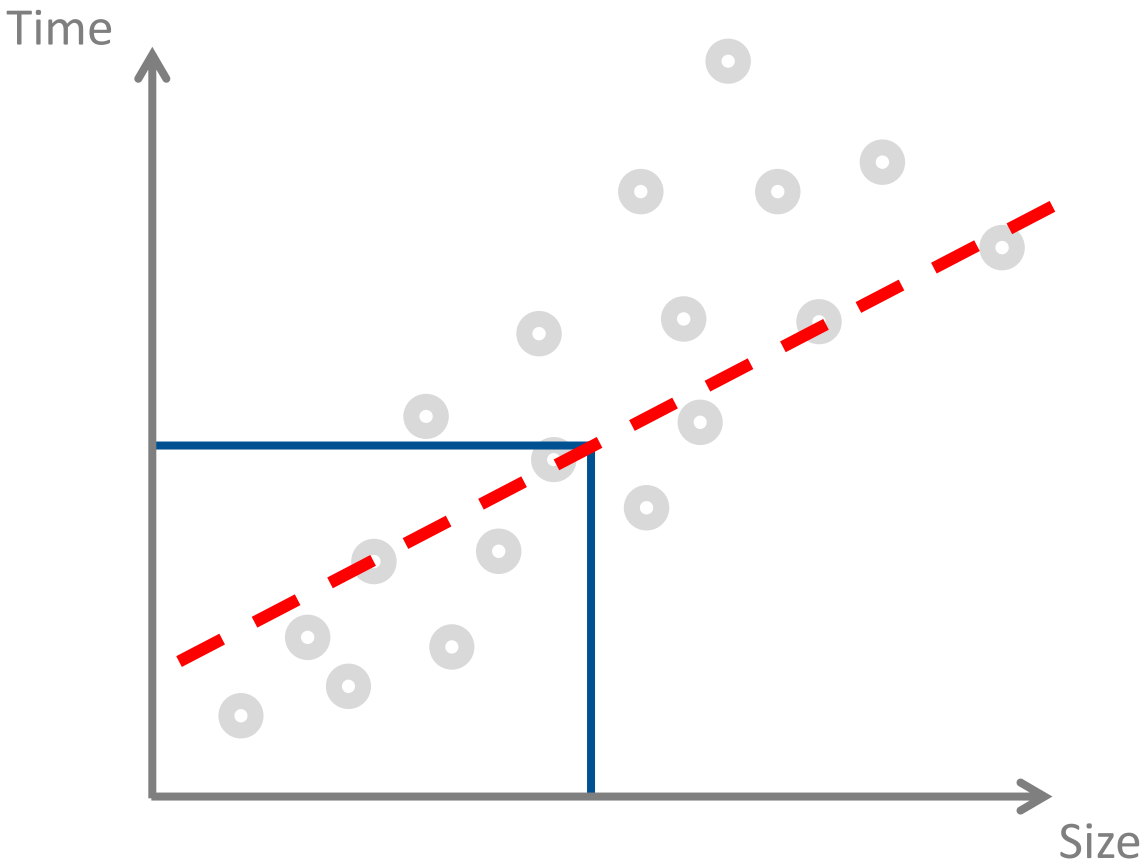
$$h(x) = \theta_0 + \theta_1 x$$



Regression

Size	Time to download
x	y
0,2 Go	2 min
1,0 Go	60 min
4,5 Go	45 min
10,0 Go	120 min

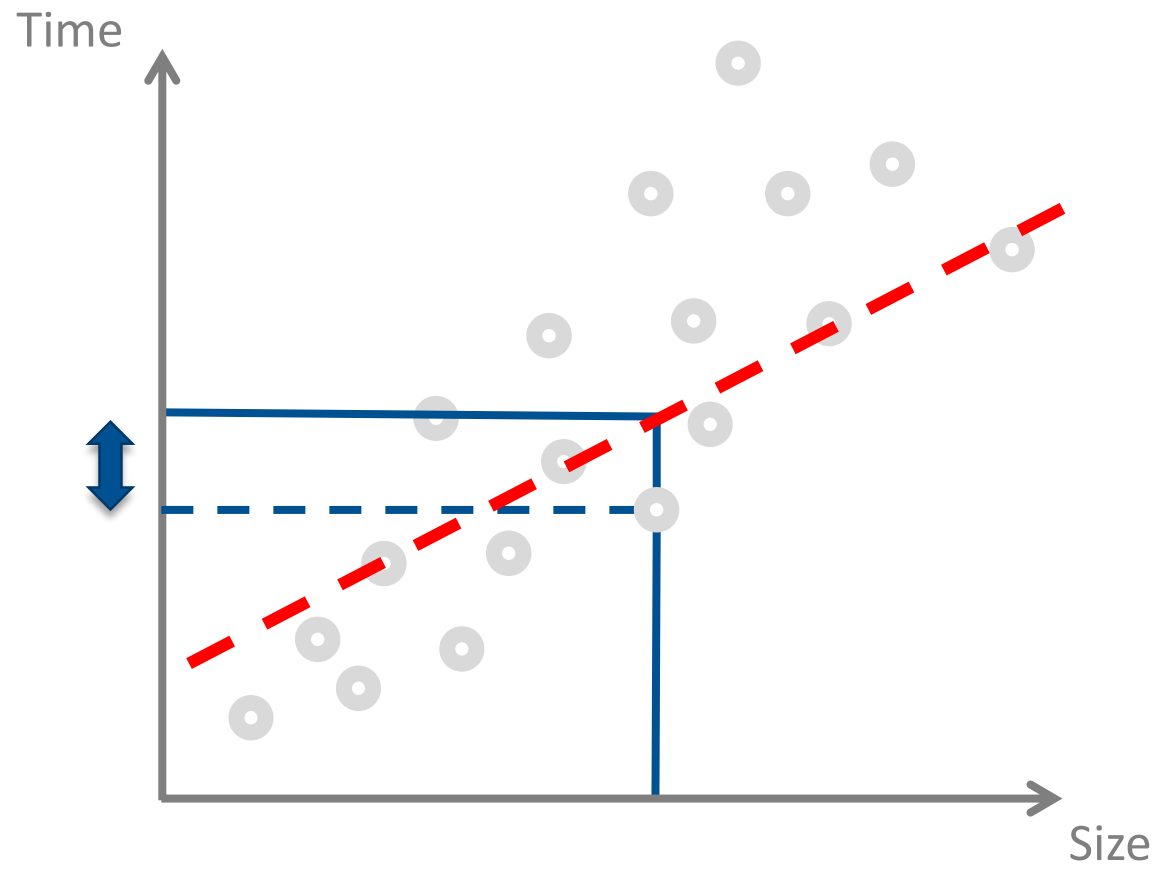
$$h(x) = \theta_0 + \theta_1 x$$



Regression

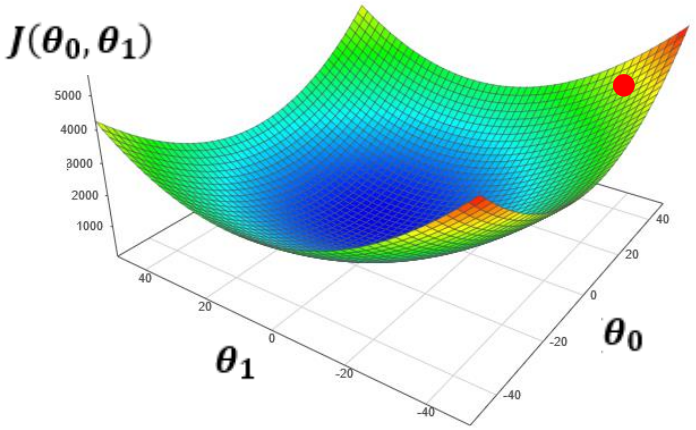
Size	Time to download
x	y
0,2 Go	2 min
1,0 Go	60 min
4,5 Go	45 min
10,0 Go	120 min

$$h(x) = \theta_0 + \theta_1 x$$

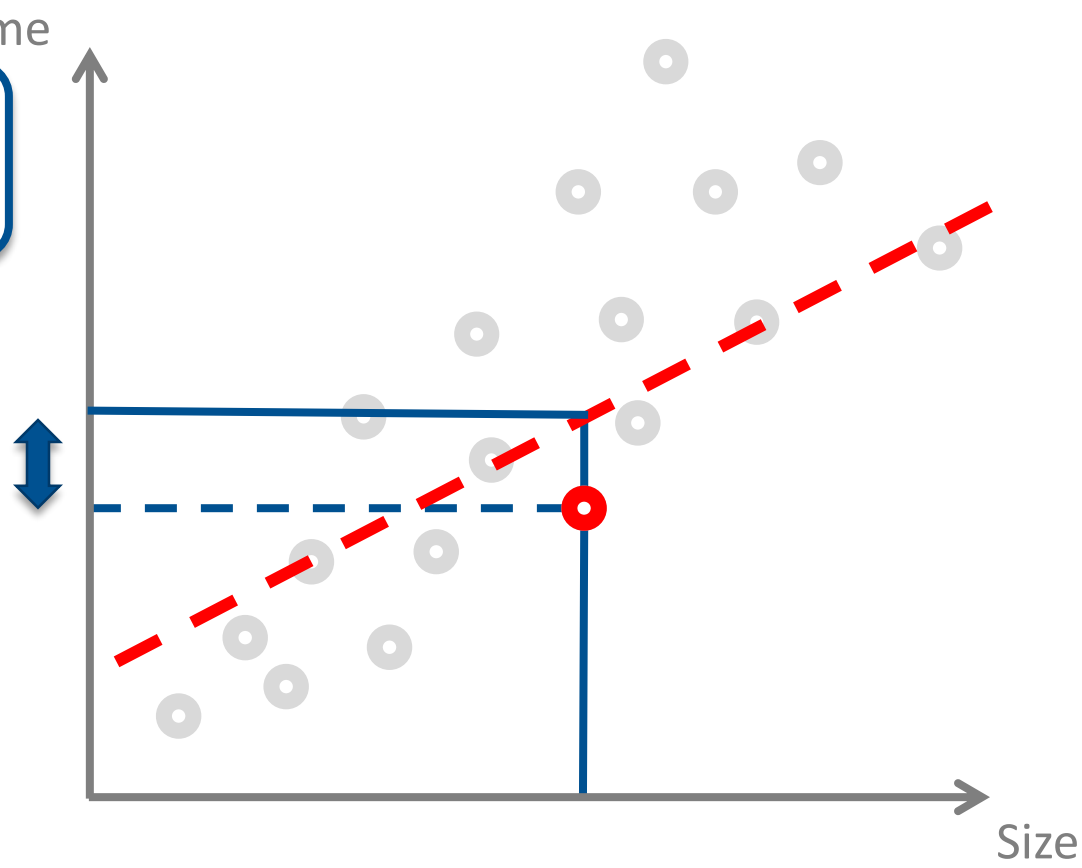


Regression

$$J(\theta_0, \theta_1) = mean\ error(h(x), y)$$

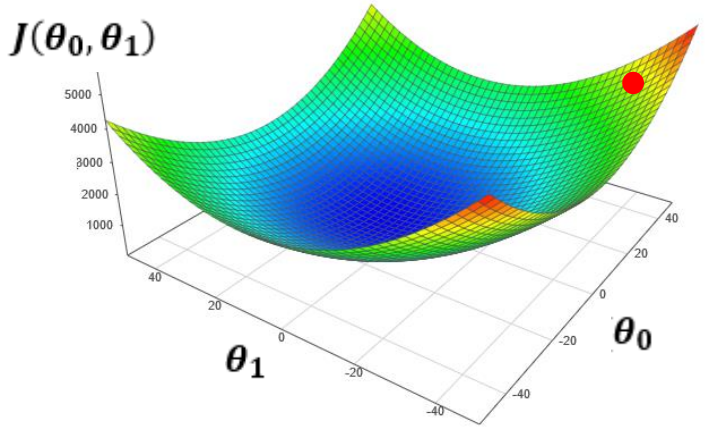


$$h(x) = \theta_0 + \theta_1 x$$

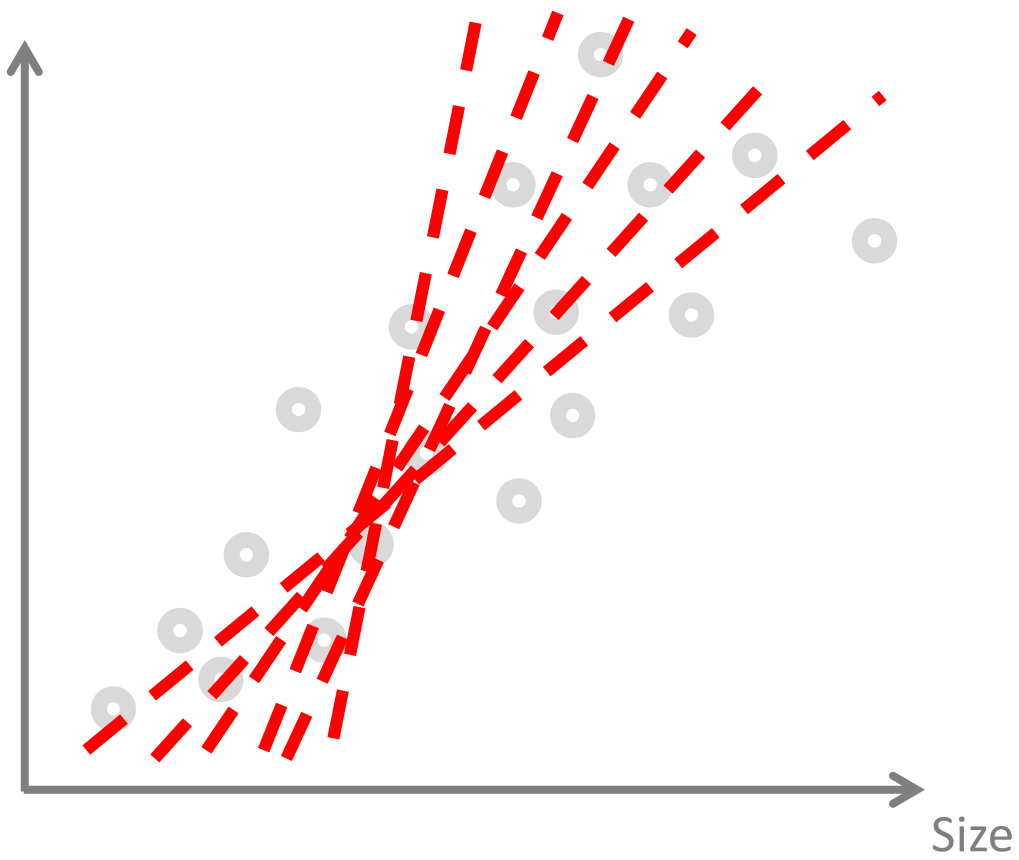


Regression - Convergence

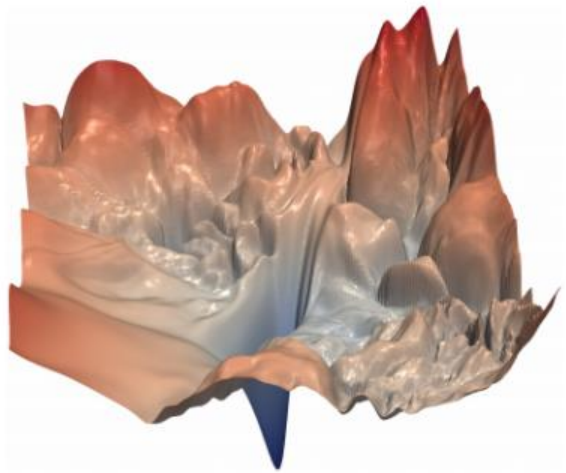
$$J(\theta_0, \theta_1) = mean\ error(h(x), y)$$



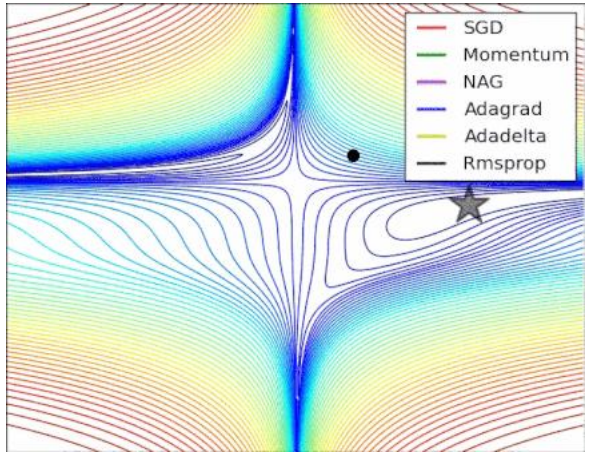
$$h(x) = \theta_0 + \theta_1 x$$



Regression - Convergence



Mean error example on a function with 2 variables



- Developing a DL algorithm is about **seeking to converge towards the optimum** of a function (maximum likelihood, minimum empirical error, etc).
- We use an **optimizer** :
  - A training data is presented multiple times to the AI which will **make numerous mistakes!**
  - Indeed, the goal of the optimizer is **NOT to have zero error for one example but to have the lowest possible error** (in "total") on all examples (especially in batch learning).
- Today, we use **the gradient descent method** (after years of research) which consists of :
  - Calculate the gradient of the cost function (the derivative of the "error" with respect to the model parameters that can be updated)
  - Then update the parameters (such as the weights of the neural network) in the direction that will decrease the error
  - We stop the learning when the mean error is no longer decreasing

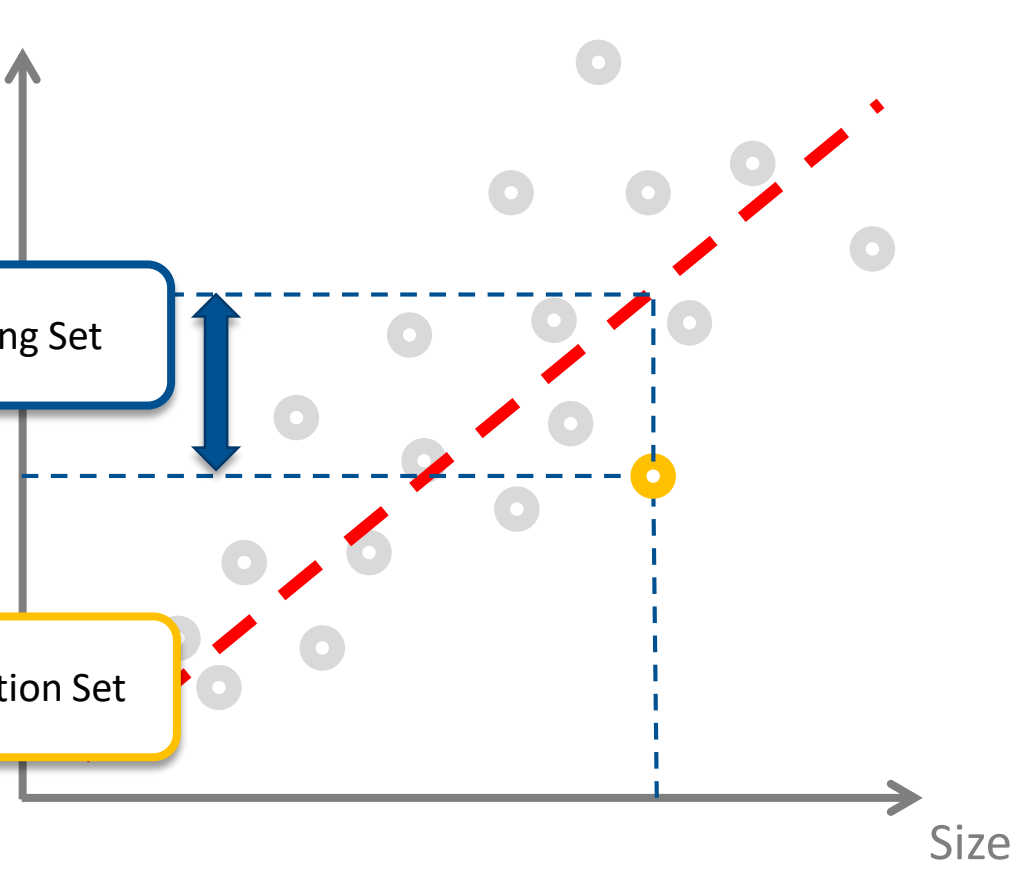
Regression

Size	Time
x	y
0,2 Go	2 min
1,0 Go	60 min
4,5 Go	45 min
10,0 Go	120 min
0,5 Go	20 min

Training Set

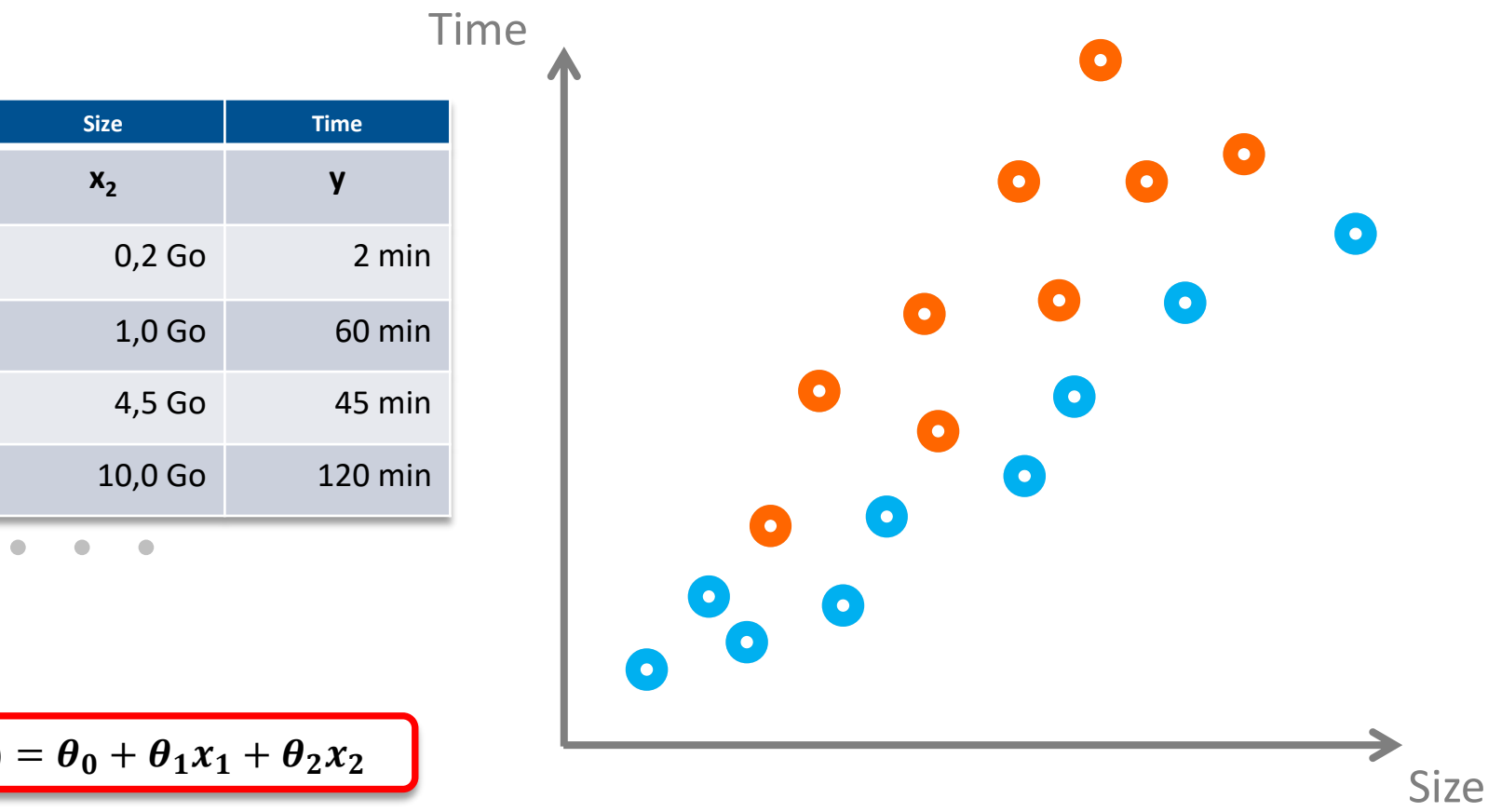
Validation Set

$$h(x) = \theta_0 + \theta_1 x$$



Regression

Server	Size	Time
x_1	x_2	y
A	0,2 Go	2 min
B	1,0 Go	60 min
B	4,5 Go	45 min
A	10,0 Go	120 min

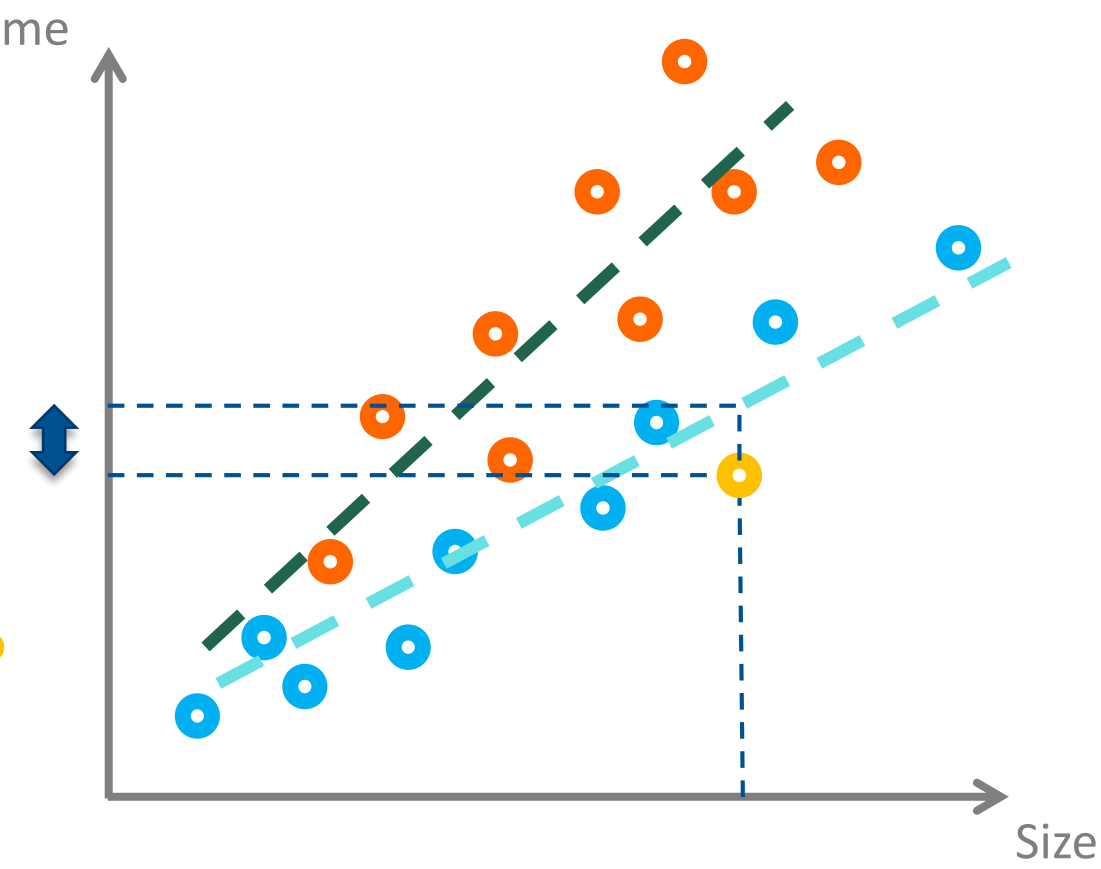


$$h(x_1, x_2) = \theta_0 + \theta_1 x_1 + \theta_2 x_2$$

Regression

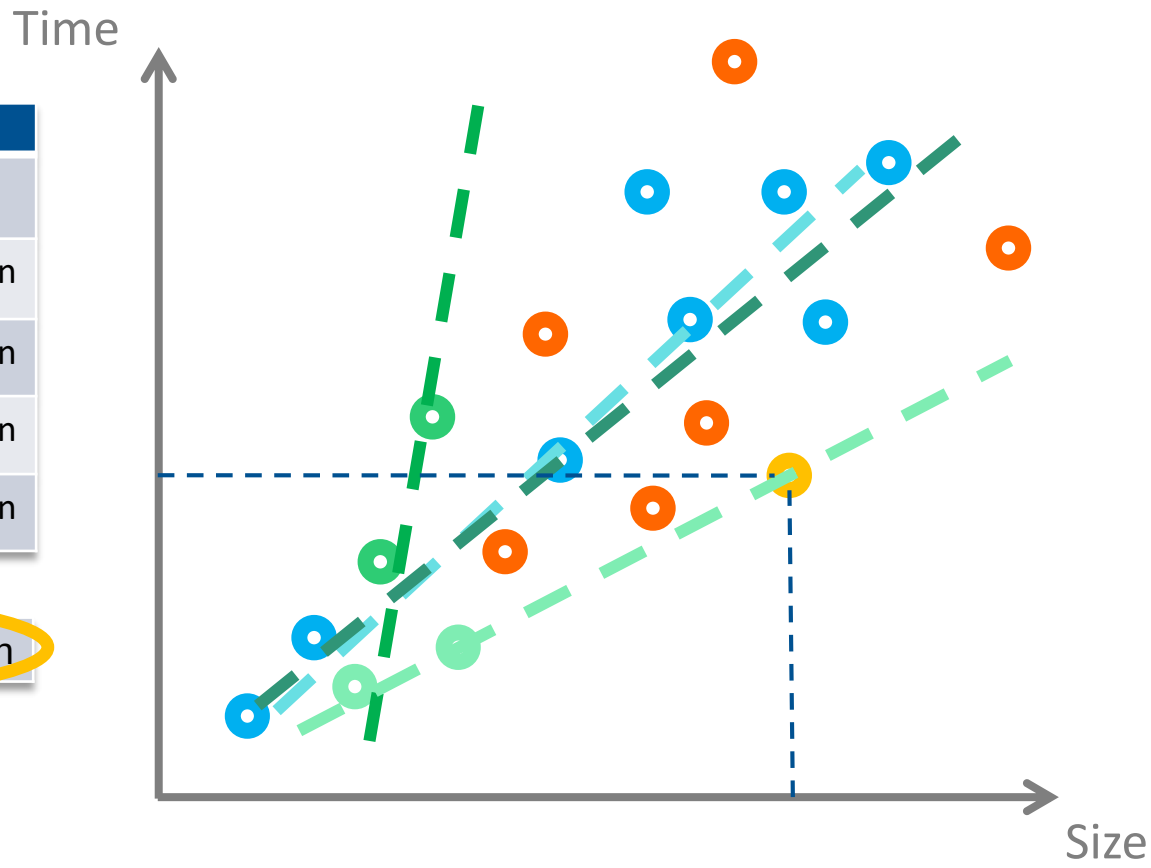
Server	Size	Time
x_1	x_2	y
A	0,2 Go	2 min
B	1,0 Go	60 min
B	4,5 Go	45 min
A	10,0 Go	120 min
B	0,5 Go	20 min

$$h(x_1, x_2) = \theta_0 + \theta_1 x_1 + \theta_2 x_2$$



Regression

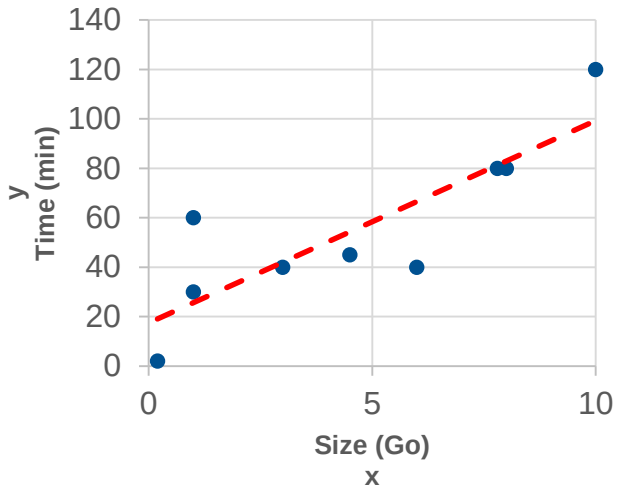
Jacket color	Size	Time
x_1	x_2	y
Rouge	0,2 Go	2 min
Bleu	1,0 Go	60 min
Orange	4,5 Go	45 min
Vert	10,0 Go	120 min
Orange	0,5 Go	20 min



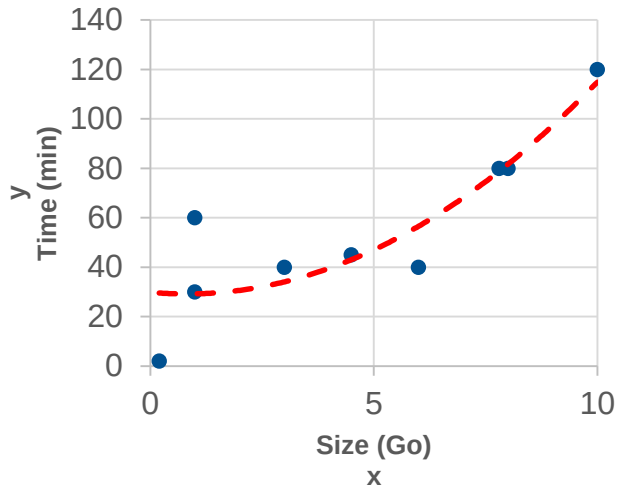
Regression

$$J = mean\ error(h(x), y)$$

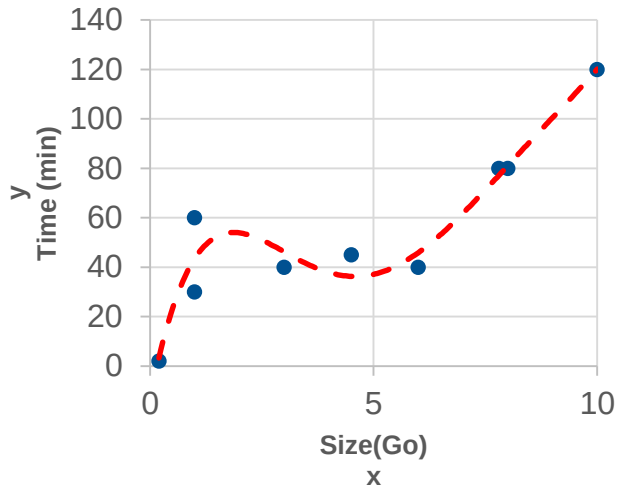
$$h(x) = \theta_0 + \theta_1 x$$



$$h(x) = \theta_0 + \theta_1 x + \theta_2 x^2$$

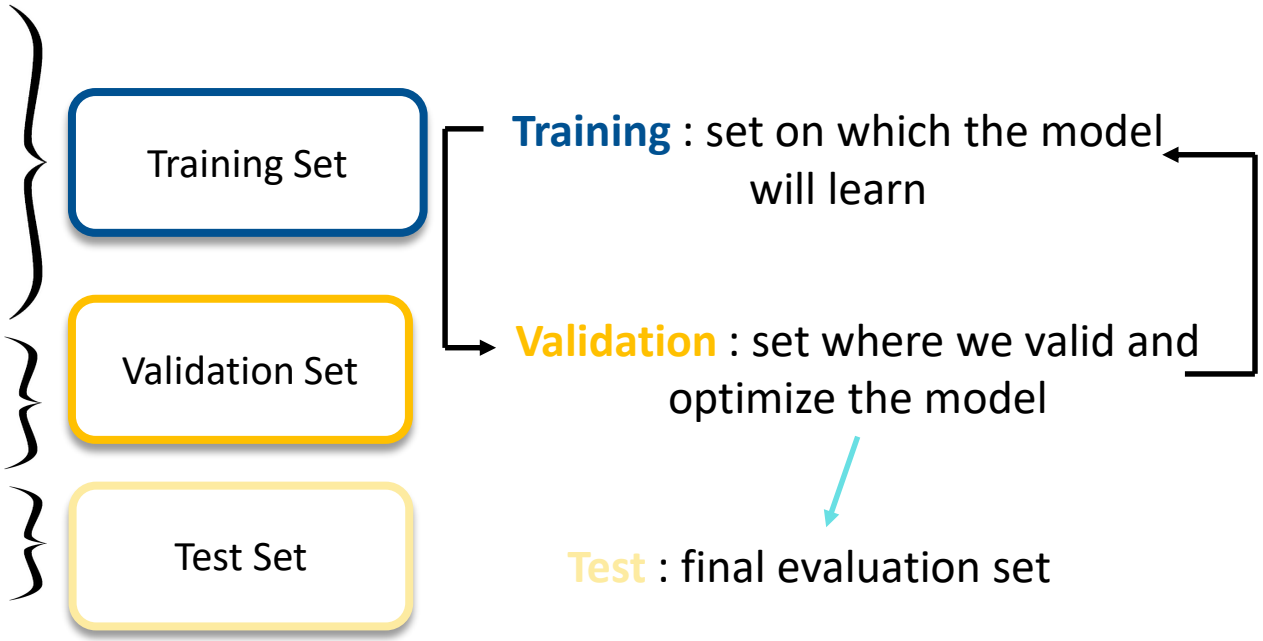


$$h(x) = \theta_0 + \theta_1 x + \theta_2 x^2 + \theta_3 x^3 + \theta_4 x^4 + \cdots$$



Regression

Time	Age of the data	Time
x_1	x_2	y
0,2 Go	6 mois	2 min
1,0 Go	24 mois	60 min
4,5 Go	1 mois	45 min
10,0 Go	12 mois	120 min
0,1 Go	0,5 mois	0,1 min
5 Go	1 mois	50 min
9 Go	10 mois	110 min
2 Go	2 mois	120 min
0,5 Go	12 mois	6 min



Regression – When to stop

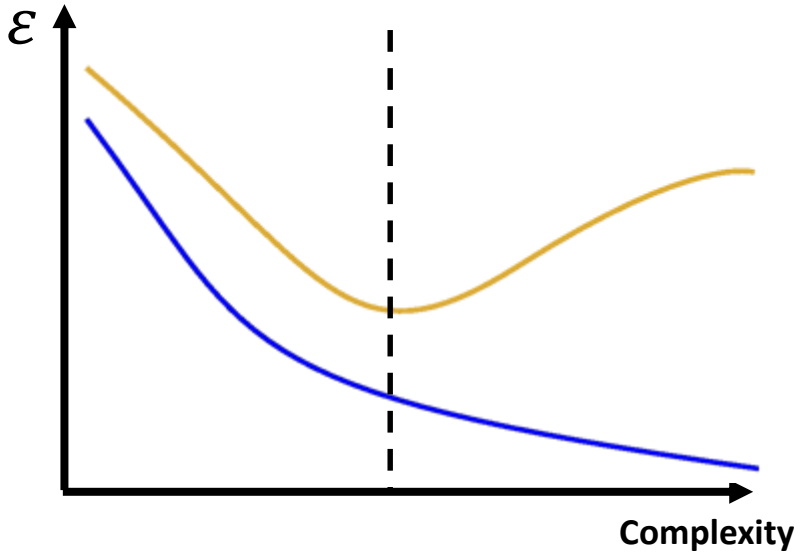
- **Do not want to minize at all cost the cost function**
- Otherwise, the AI may not be able to generalize
- Bias-variance tradeoff

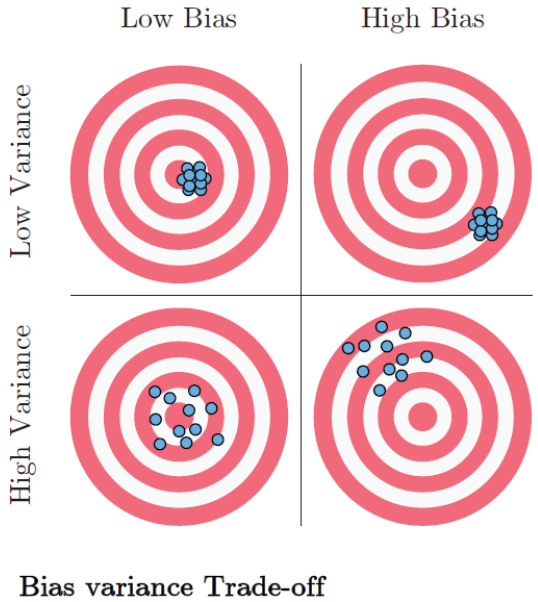
Size	Age	Time
x_1	X_2	Y
0,2 Go	6 mois	2 min
1,0 Go	24 mois	60 min
4,5 Go	1 mois	45 min
10,0 Go	12 mois	120 min
0,1 Go	0,5 mois	0,1 min
5 Go	1 mois	50 min
9 Go	10 mois	110 min
2 Go	2 mois	120 min
0,5 Go	12 mois	6 min

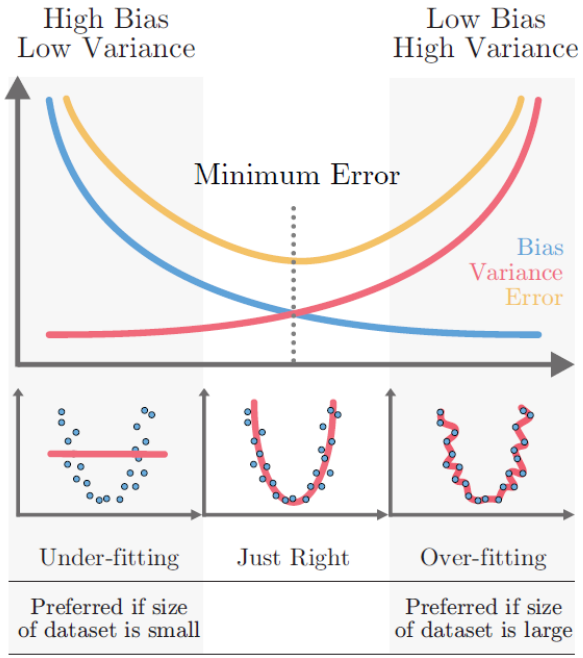
Training Set

Validation Set

Bias / Variance Tradeoff







Ml Process

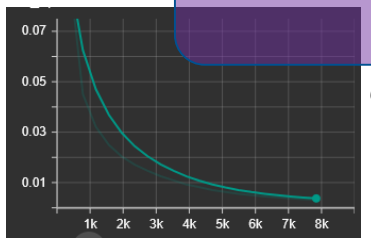
Need definition

DATA

Quantity
Quality
Représentativity

Are the data representative ?
Is the scenario well defined ?

Validation



ϵ : Validation error

Can the model read the data ?

Are the hypothesis of the model well defined ?

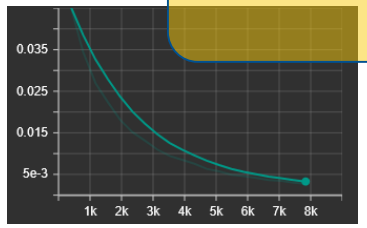
MODEL

Hypothesis
Generalization
Outputs

Is it the best cost function ?

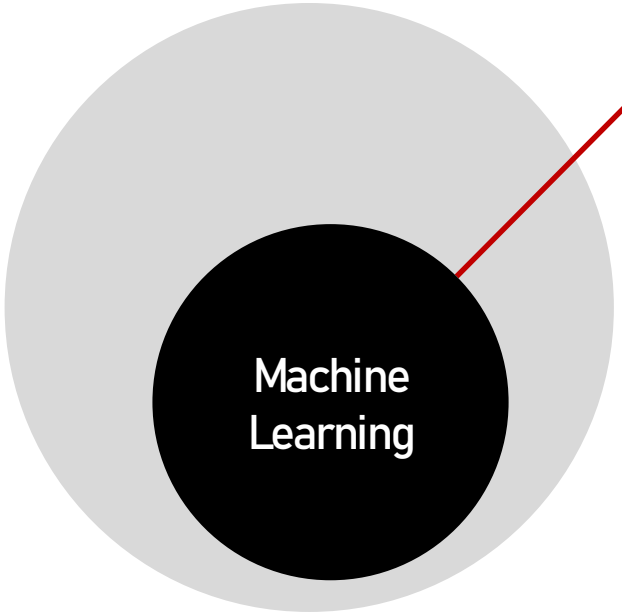
Are the evaluation metrics good ?

Apprentissage



ϵ : Learning error

Classification



Supervised

Regression

- Linear Regression
- Polynomial Regression
- Ridge/Lasso Regression

Classification

- Logistic regression
- SVM
- K-NN
- Decision Tree
- Random Forest
- Naïves Bayes

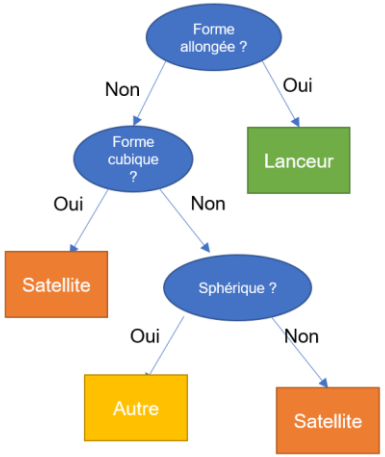
discrete

Classification and Clustering...

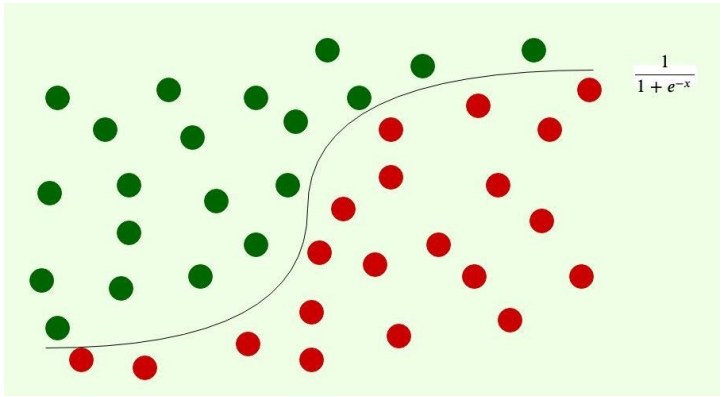
The term "clustering" is one of the curious word of the data scientist; it refers to the concept of unsupervised classification

Supervised classification (Classification)

- Needs to already have partitionned data
- The idea is to teach the machine learning model the underlying distribution of the data so that it can assign a new data point to its appropriate class. This is done by training the model on a labeled dataset, where the target variable is known for each data point



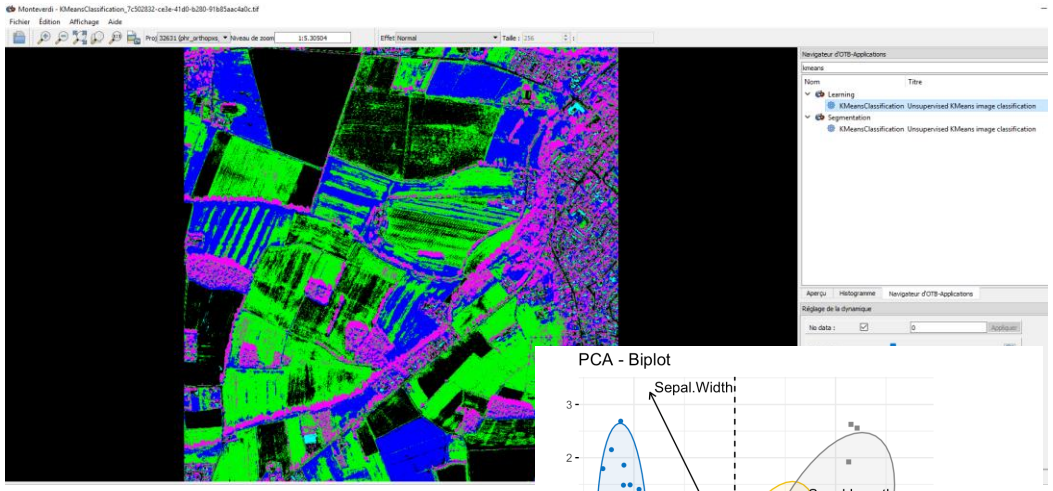
Random Forest



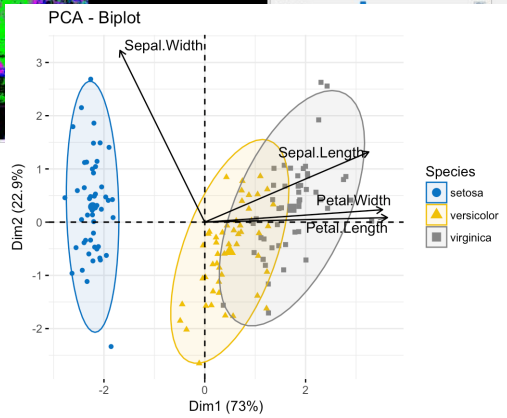
Logistic Regression

Unsupervised classification (Clustering)

- Clustering is about finding a "natural" structure in the data, since no target typology is provided prior to the algorithm.



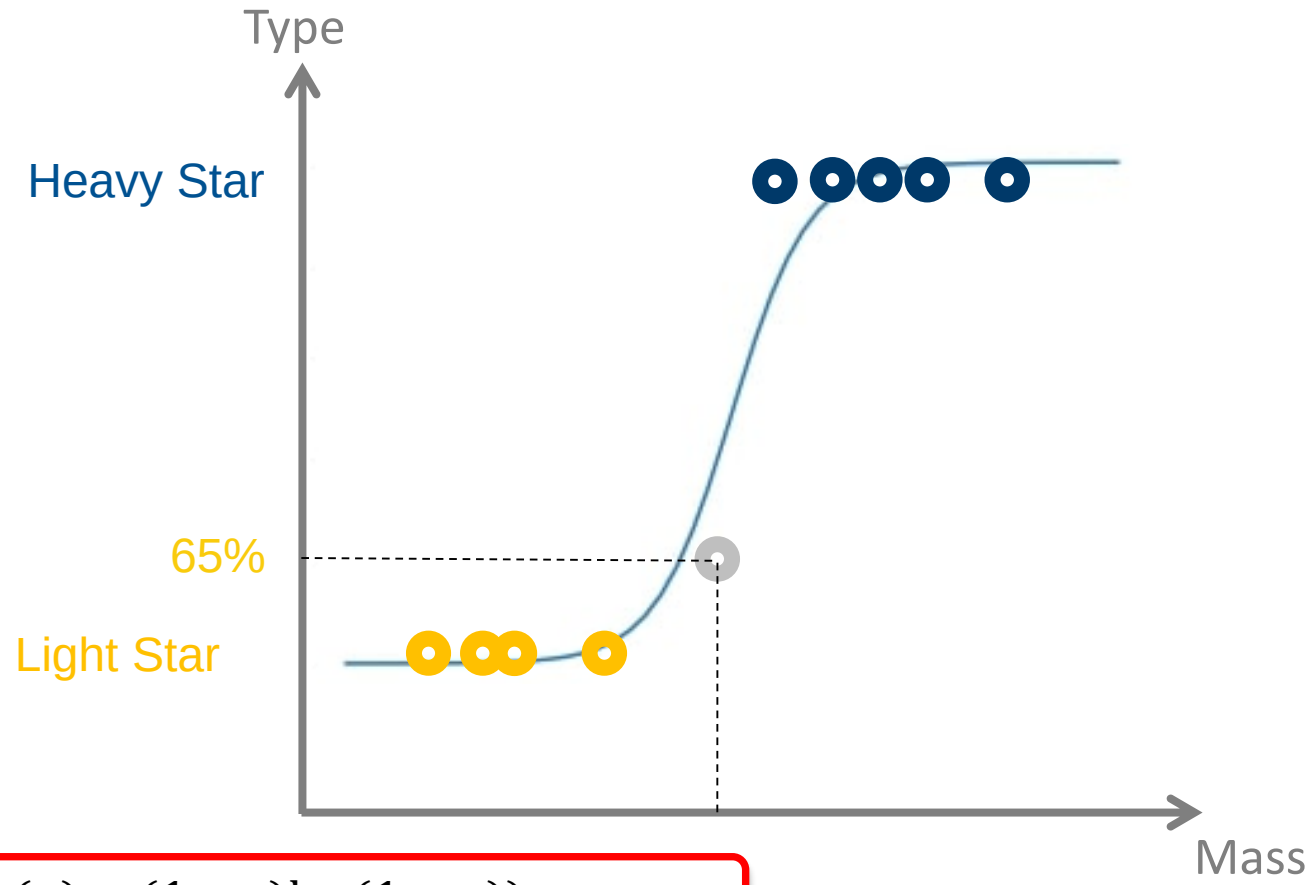
Clustering



PCA

Logistic Regression

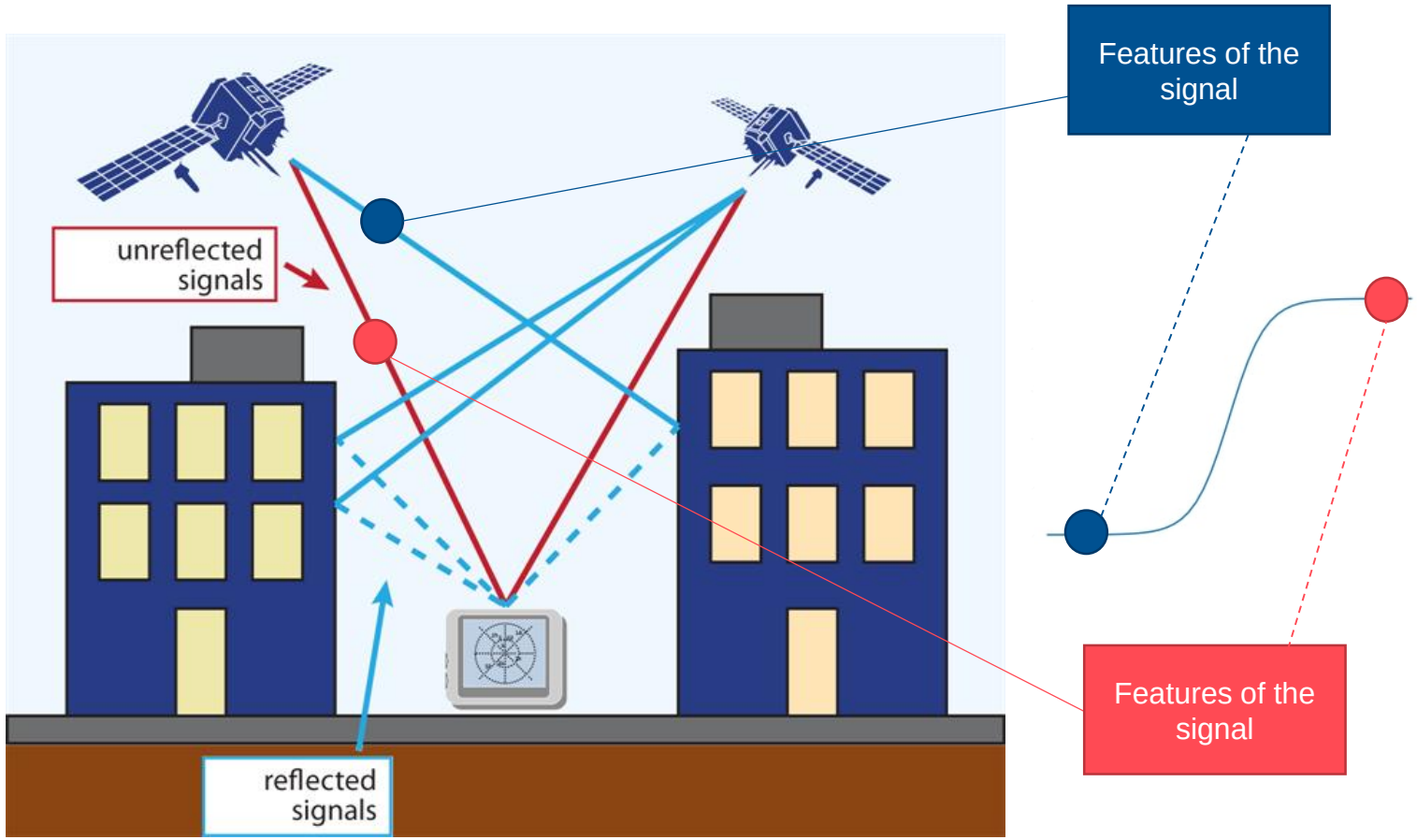
Name	Mass ($M_\odot$)	Star type
Sigma Draconis	0.85	Orange
107 Piscium	0.86	Orange
Epsilon Eridani	0.82	Orange
Omega Fornacis	3.42	Bleue
KIC 7760680	3.25	Bleue
18 Tauri	3.34	Bleue
HD 2071	3.69	Bleue



$$h(x) = -(y log(p) + (1 - y)\log(1 - p))$$

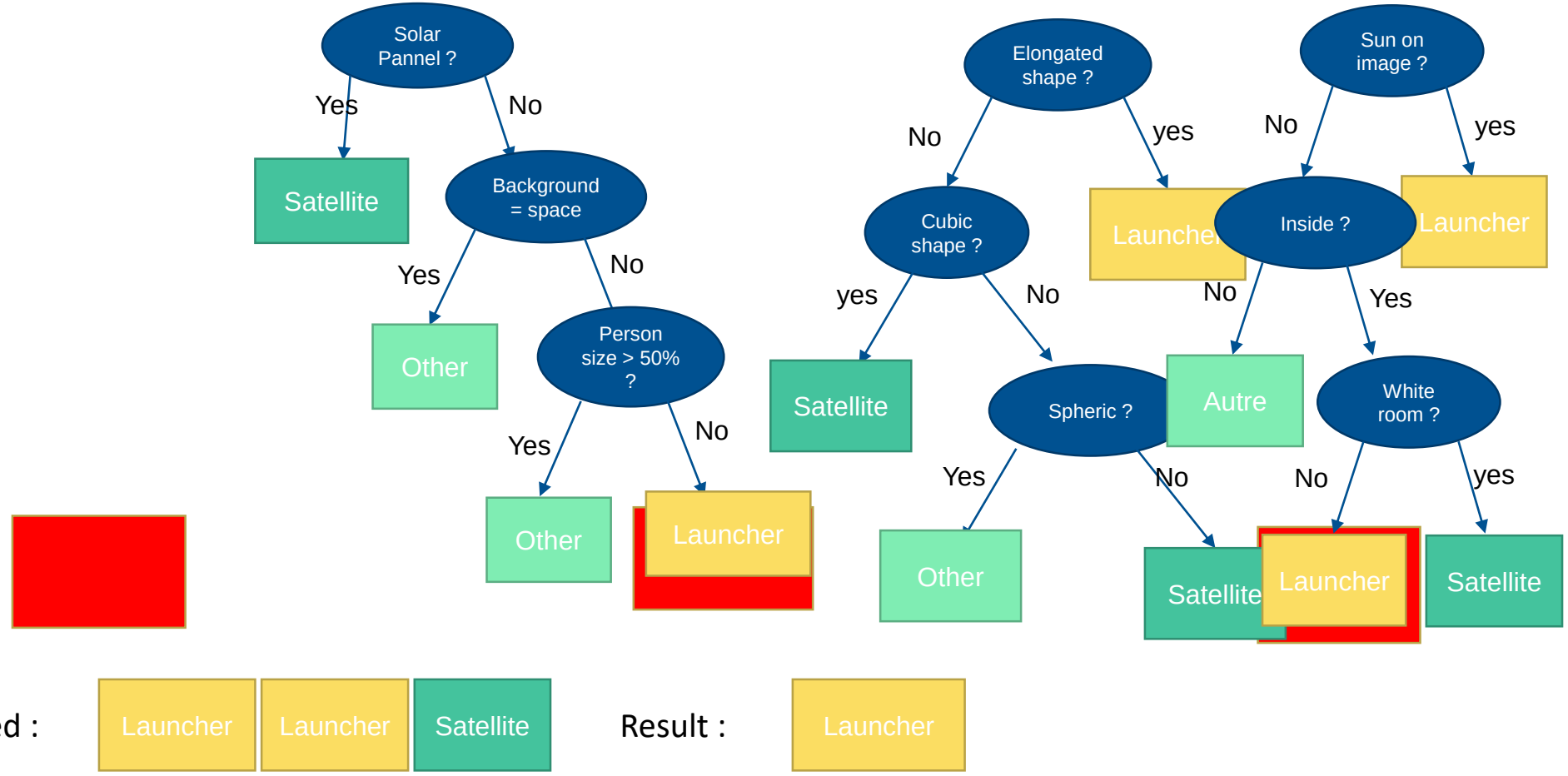
Logistic Regression – CNES example

Data	GNSS Signal
Objective	Detect reflected signals



Supervised – Random Forest

Launcher, Satellite or something else?



Predicted : Launcher | Launcher | Satellite

Result : Launcher

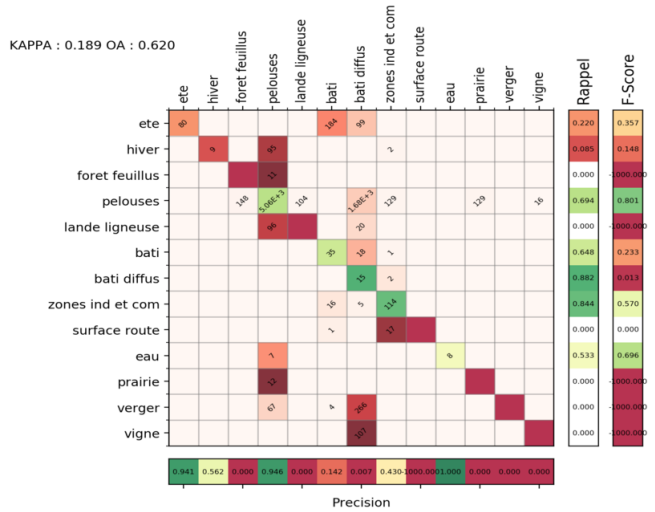
Random Forest : Boosting of weak models, each using different features

Random Forest – IOTA-2 / OSO

- Land use map on France
- Classification using Random Forest
- Sentinel 2: revisit, complete coverage and several spectral bands

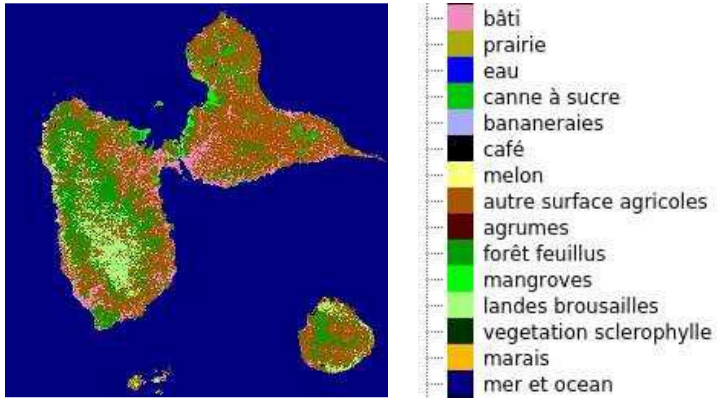
OSO sur la France métropolitaine (CESBIO)

http://osr-cesbio.ups-tlse.fr/~oso/ui-ol/S2_2017/layer.html



- Annual Summer Crops (ASC)
- Annual Winter Crops (AWC)
- Broad-leaved Forests (BLF)
- Coniferous Forests (COF)
- Natual Grasslands (NGL)
- Woody Moorlands (WOM)
- Continuous Urban Fabric (CUF)
- Discontinuous Urban Fabric (DUF)
- Industrial and Commercial Units (ICU)
- Road Surfaces (RDF)
- Bare Rock (BRO)
- Beaches, dunes and sand (BDS)
- Water Bodies (WAT)
- Glacier and perpetual snow (GPS)
- Intensive Grasslands (IGL)
- Orchards (ORC)
- Vineyards (VIN)

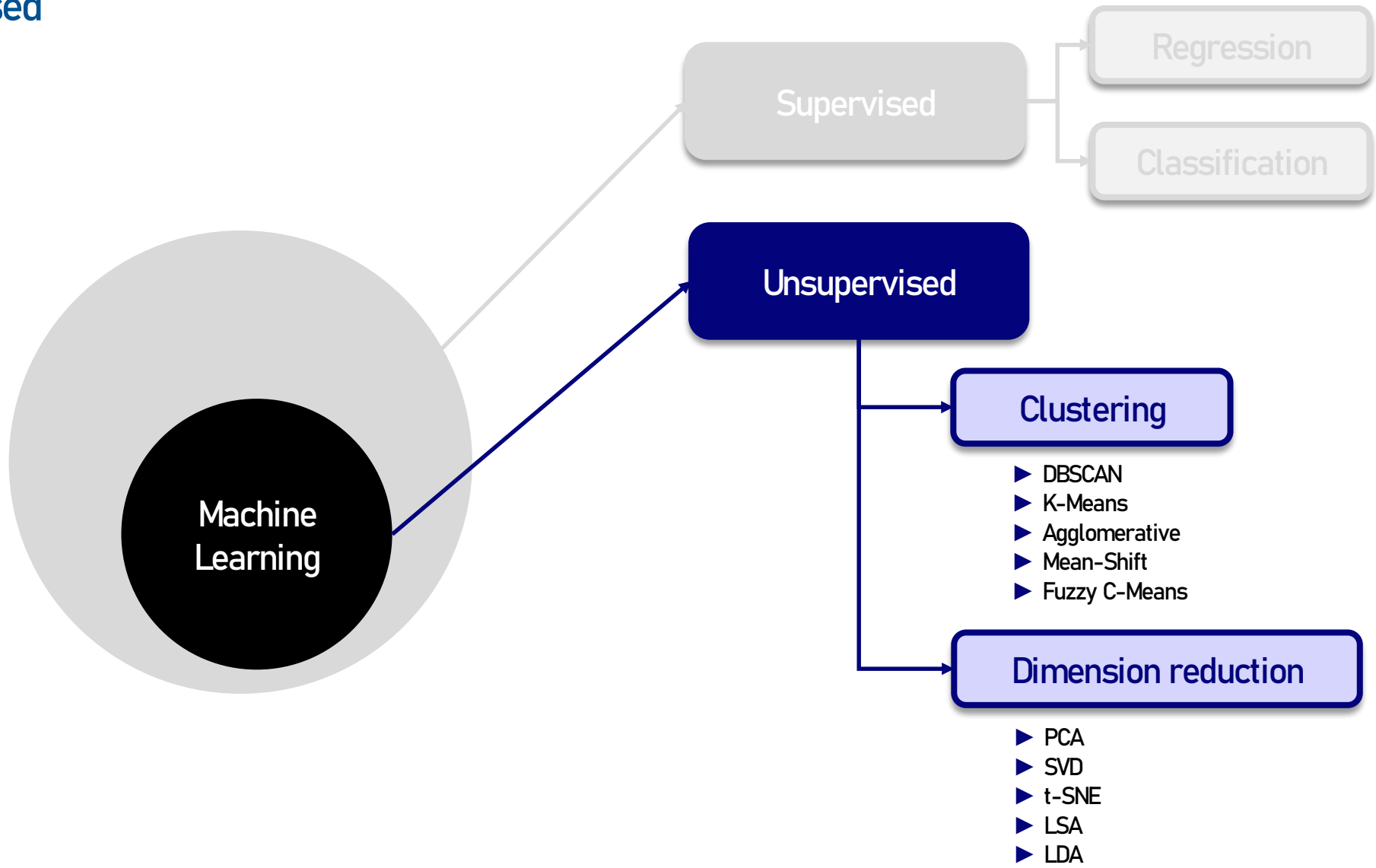
sur la Guadeloupe (CNES)



Agriculture

Unsupervised

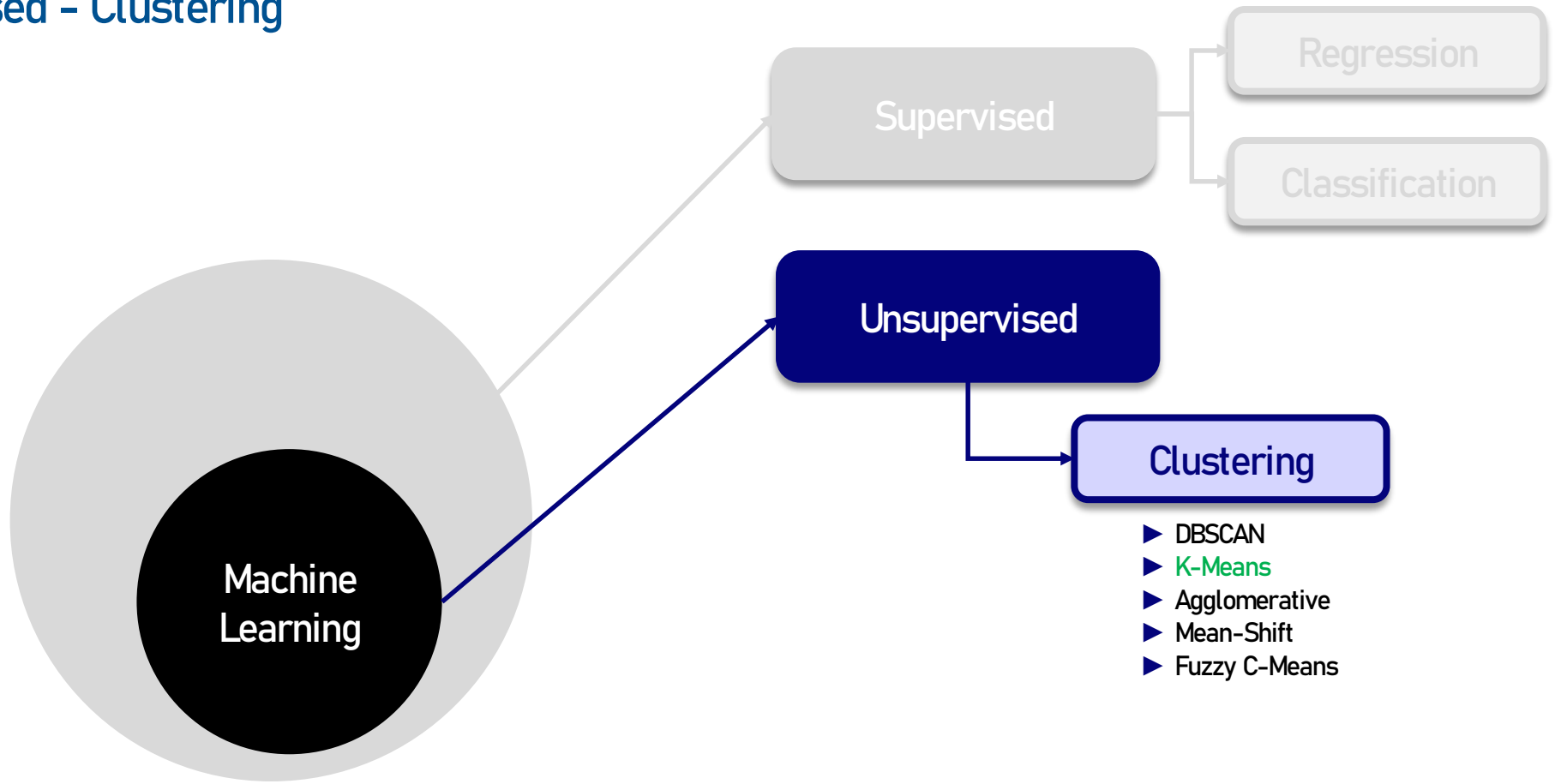
Unsupervised



- Linear Regression
- Polynomial Regression
- Ridge/lasso Regression

- Logistic Regression
- SVM
- K-NN
- Decision Tree
- Random Forest
- Naïves Bayes

Unsupervised - Clustering



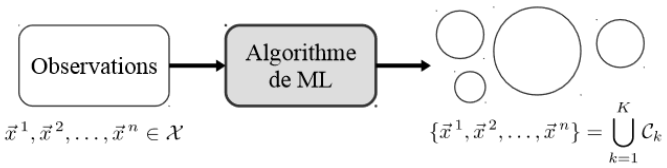
- Linear Regression
- Polynormial Regression
- Ridge/Lasso Regression

- Logistic Regression
- SVM
- K-NN
- Decision Tree
- Random Forest
- Naïves Bayes

Unsupervised - Clustering

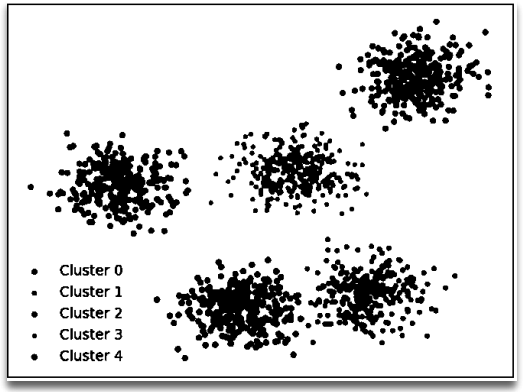
Divide data into clusters

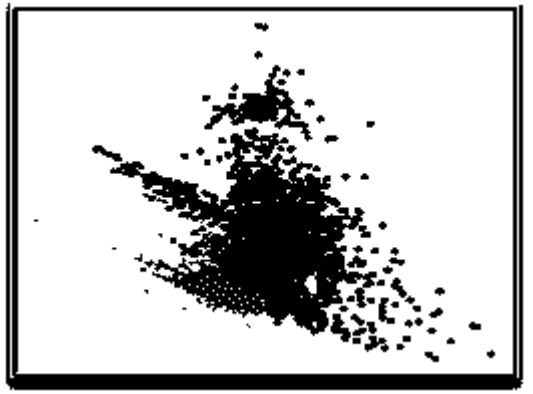
- Clustering is about searching for a "natural" structure in the data, since no target typology is provided prior to the algorithm
- The idea is to determine classes that are as homogeneous as possible while being as distinct as possible from each other.
- It becomes necessary to determine a measure of separability between classes and similarity (or dissimilarity) between individuals.



Dataset

Unsupervised





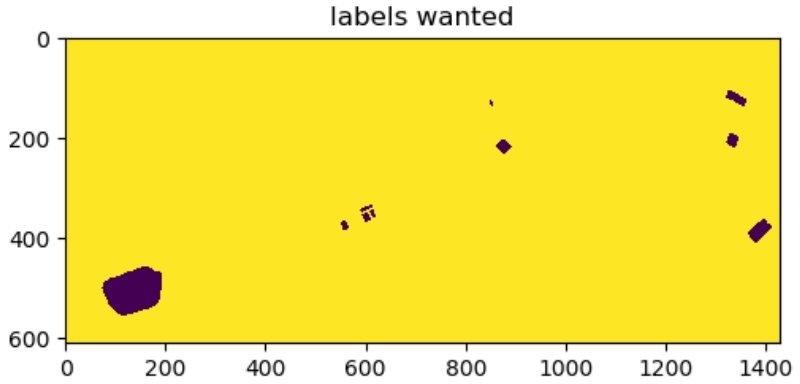
$$\begin{pmatrix} x_{11} & \cdots & x_{1n} \\ \vdots & \ddots & \vdots \\ x_{m1} & \cdots & x_{mn} \end{pmatrix}$$

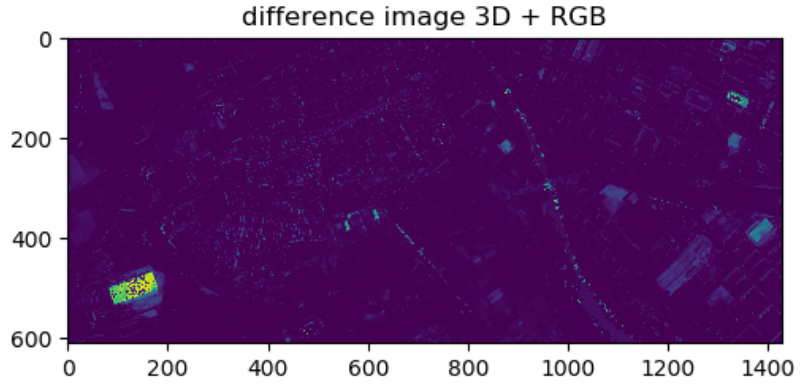
How?

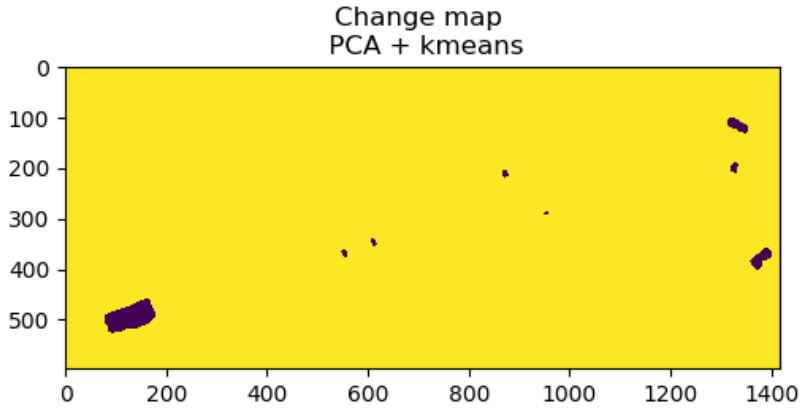
- By mesure of similarity (distances, densities...)
- Can be by shifting the features spaces into another one

Unsupervised – Clustering – K-Means – CNES example

Data	Images
Objective	Determine 3D change on a urban landscape

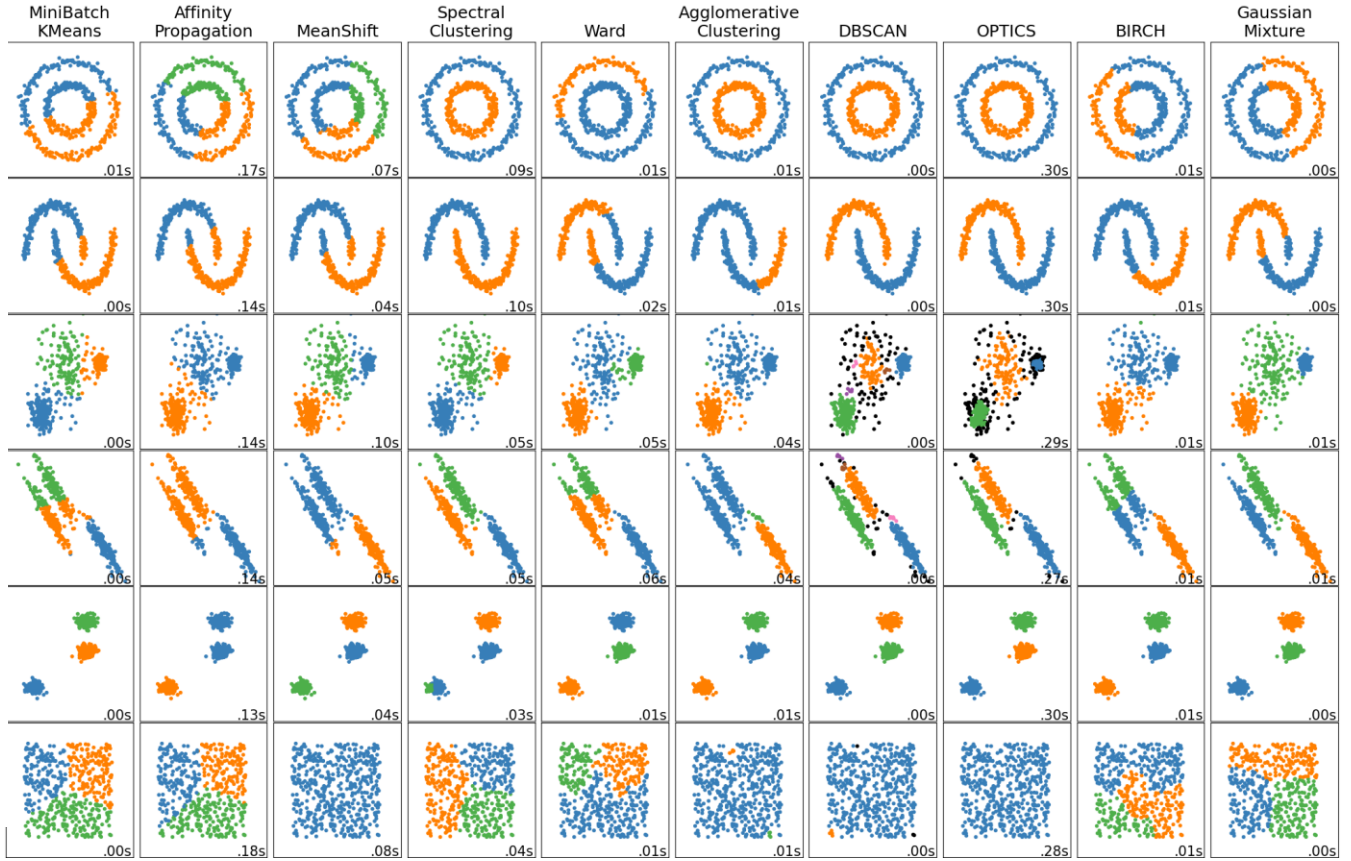






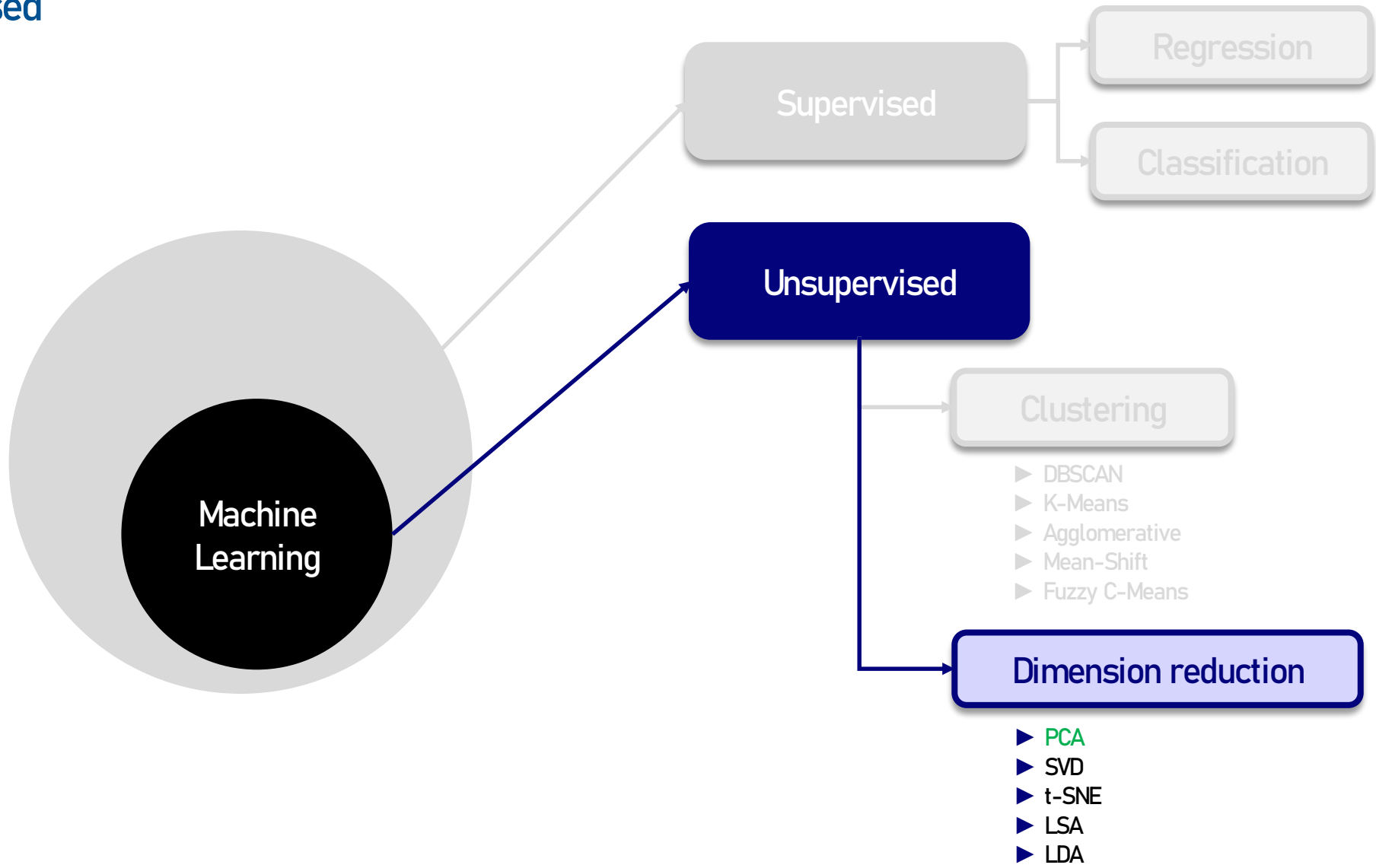
Unsupervised – Clustering – Panorama of K-medoids methods

Method name	Parameters	Scalability	Usecase	Geometry (metric used)
K-Means	number of clusters	Very large `n_samples`, medium `n_clusters` with MiniBatch code	General-purpose, even cluster size, flat geometry, not too many clusters, inductive	Distances between points
Affinity propagation	damping, sample preference	Not scalable with n_samples	Many clusters, uneven cluster size, non-flat geometry, inductive	Graph distance (e.g. nearest-neighbor graph)
Mean-shift	bandwidth	Not scalable with `n_samples`	Many clusters, uneven cluster size, non-flat geometry, inductive	Distances between points
Spectral clustering	number of clusters	Medium `n_samples`, small `n_clusters`	Few clusters, even cluster size, non-flat geometry, transductive	Graph distance (e.g. nearest-neighbor graph)
Ward hierarchical clustering	number of clusters or distance threshold	Large `n_samples` and `n_clusters`	Many clusters, possibly connectivity constraints, transductive	Distances between points
Agglomerative clustering	number of clusters or distance threshold, linkage type, distance	Large `n_samples` and `n_clusters`	Many clusters, possibly connectivity constraints, non Euclidean distances, transductive	Any pairwise distance
DBSCAN	neighborhood size	Very large `n_samples`, medium `n_clusters`	Non-flat geometry, uneven cluster sizes, outlier removal, transductive	Distances between nearest points
HDBSCAN	minimum cluster membership, minimum point neighbors	large `n_samples`, medium `n_clusters`	Non-flat geometry, uneven cluster sizes, outlier removal, transductive, hierarchical, variable cluster density	Distances between nearest points
OPTICS	minimum cluster membership	Very large `n_samples`, large `n_clusters`	Non-flat geometry, uneven cluster sizes, variable cluster density, outlier removal, transductive	Distances between points
Gaussian mixtures	many	Not scalable	Flat geometry, good for density estimation, inductive	Mahalanobis distances to centers
BIRCH	branching factor, threshold, optional global clusterer.	Large `n_clusters` and `n_samples`	Large dataset, outlier removal, data reduction, inductive	Euclidean distance between points
Bisecting K-Means	number of clusters	Very large `n_samples`, medium `n_clusters`	General-purpose, even cluster size, flat geometry, no empty clusters, inductive, hierarchical	Distances between points



Characteristics of different clustering algorithms on "interesting" 2D datasets

Unsupervised

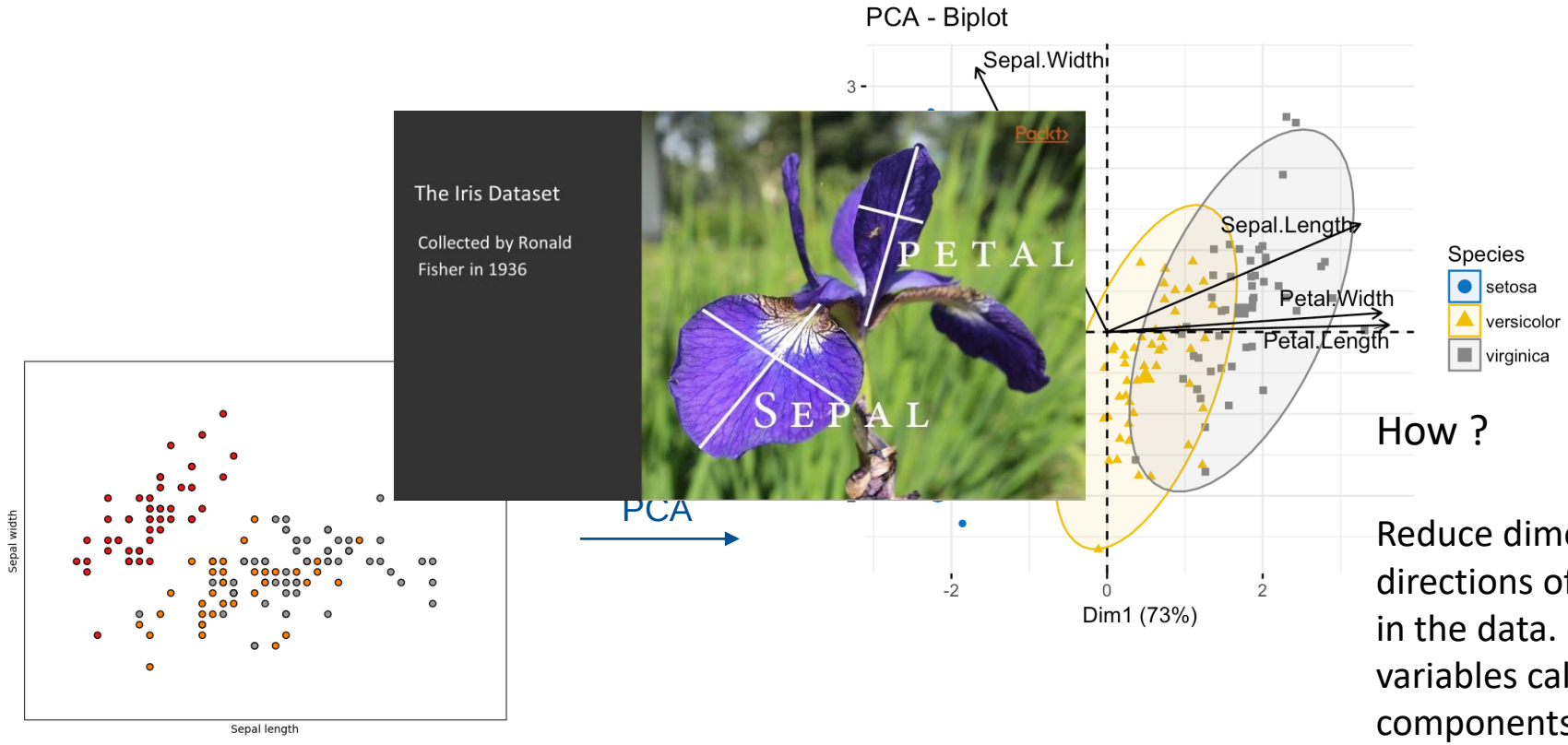


- Linear Regression
- Polynomial Regression
- Ridge/Lasso Régression

- Logistic Régression
- SVM
- K-NN
- Decision Tree
- Random Forest
- Naïves Bayes

- DBSCAN
- K-Means
- Agglomerative
- Mean-Shift
- Fuzzy C-Means

Unsupervised – Principal Component Analysis

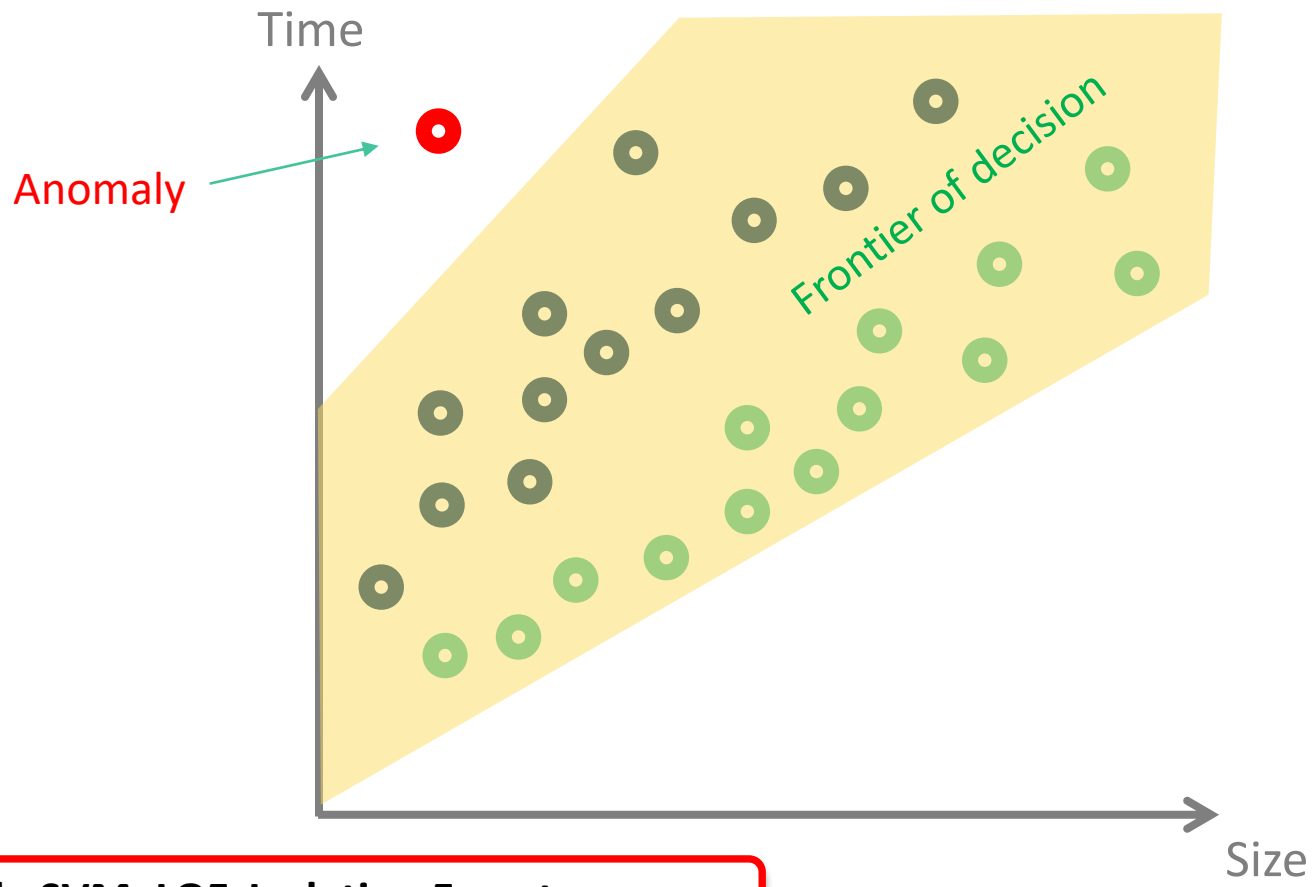


How ?

Reduce dimensionality by finding directions of maximum variance in the data. It creates new variables called principal components that capture the most significant variation in the data

Unsupervised – Anomaly Detection

Size	Time
0,2 Go	2 min
1,0 Go	60 min
15,0 Go	100 min
25,0 Go	200 min
0,5 Go	3 min
4,5 Go	45 min
10 Go	100 min

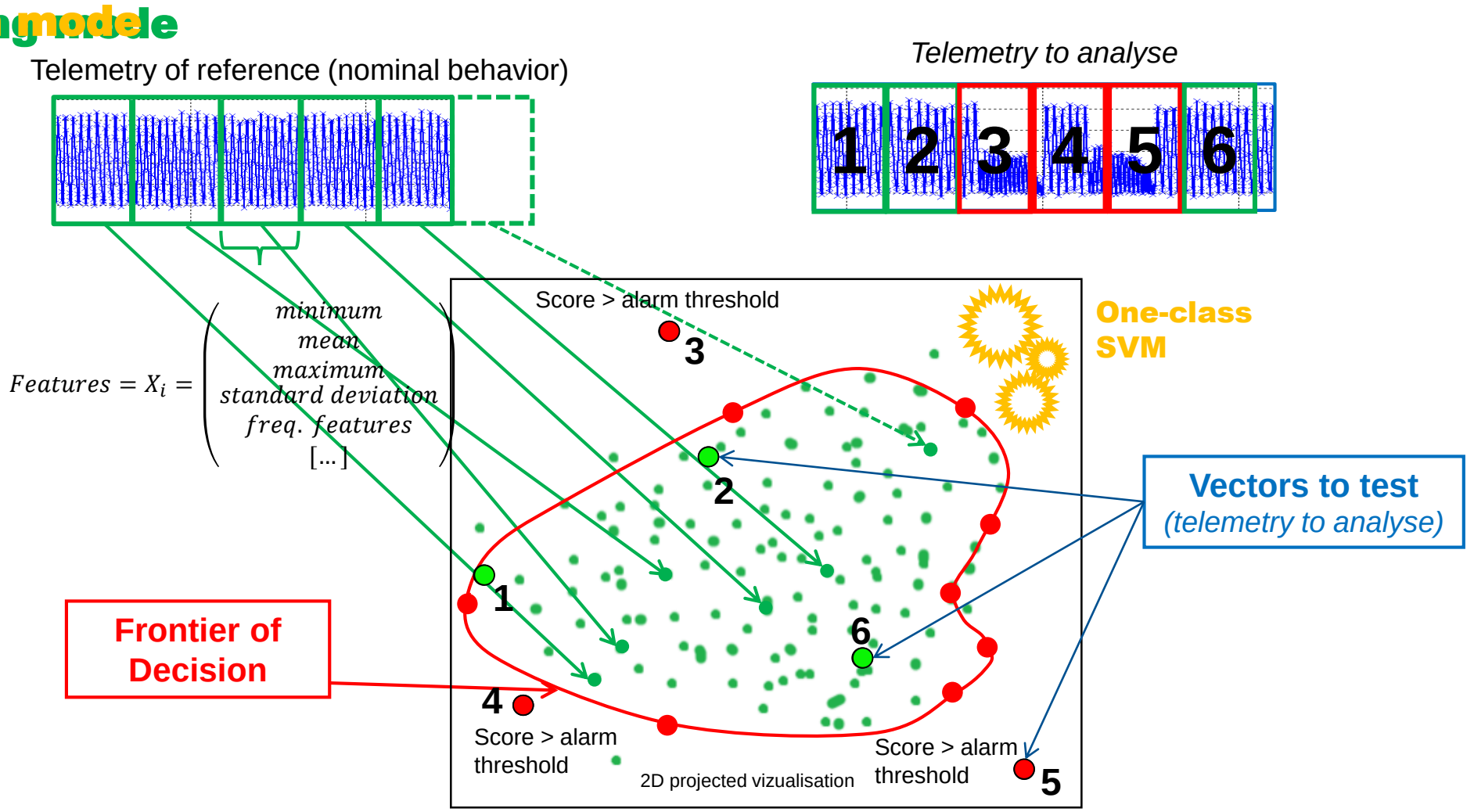


Possible model : SVM, LOF, Isolation Forest

Unsupervised – Anomaly detection – CNES example

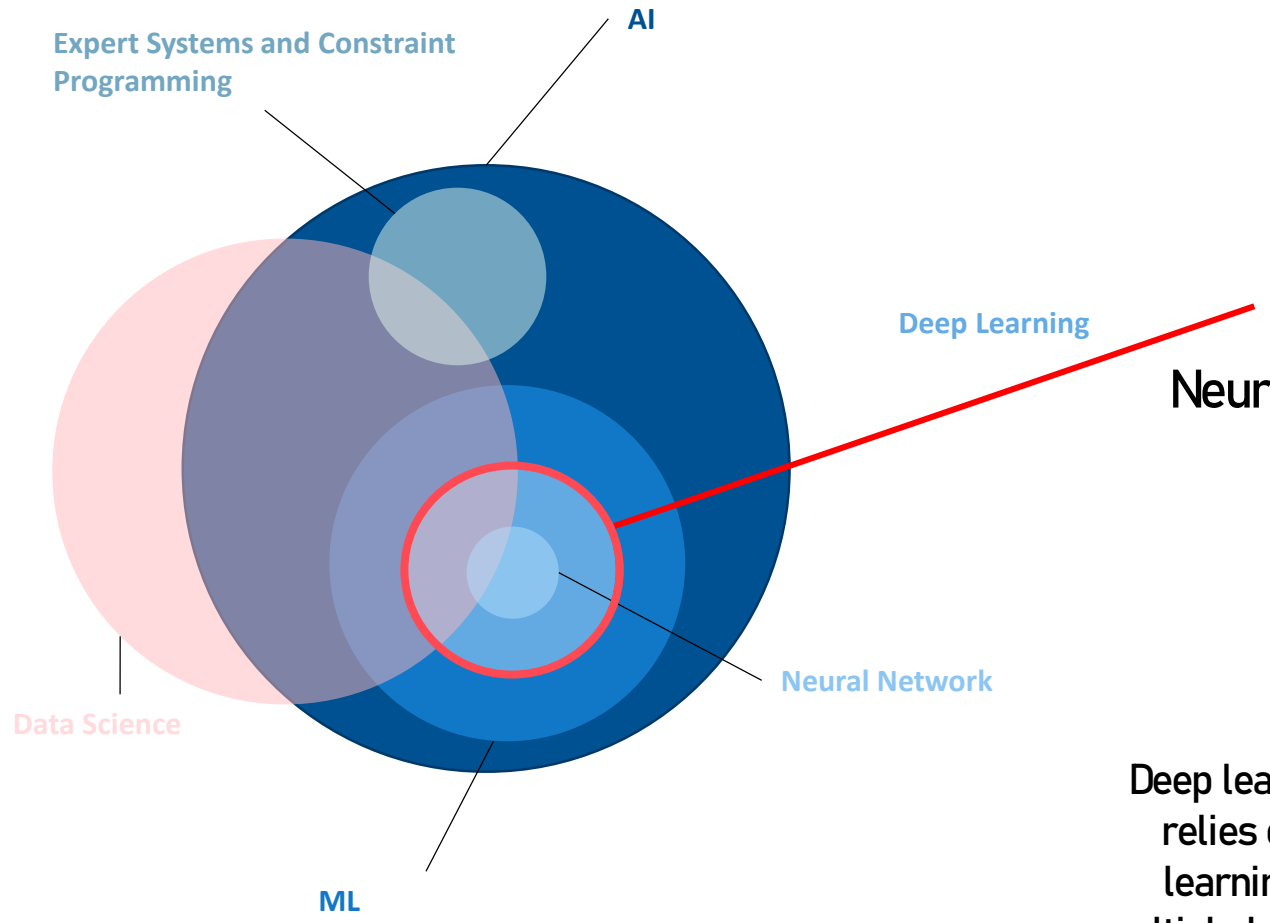
NOSTRADAMUS

- **Anomaly detection on satellite telemetry**
- **One-Class SVM algorithm**
- **Developed and licensed by CNES since 2014**
- **Used on different operational plaforms**
- **Weak signal detection allows for the detection of early signs of a weak signal that may not be detected by traditional monitoring methods**



Deep Learning

Artificial Intelligence, Machine Learning and Deep Learning



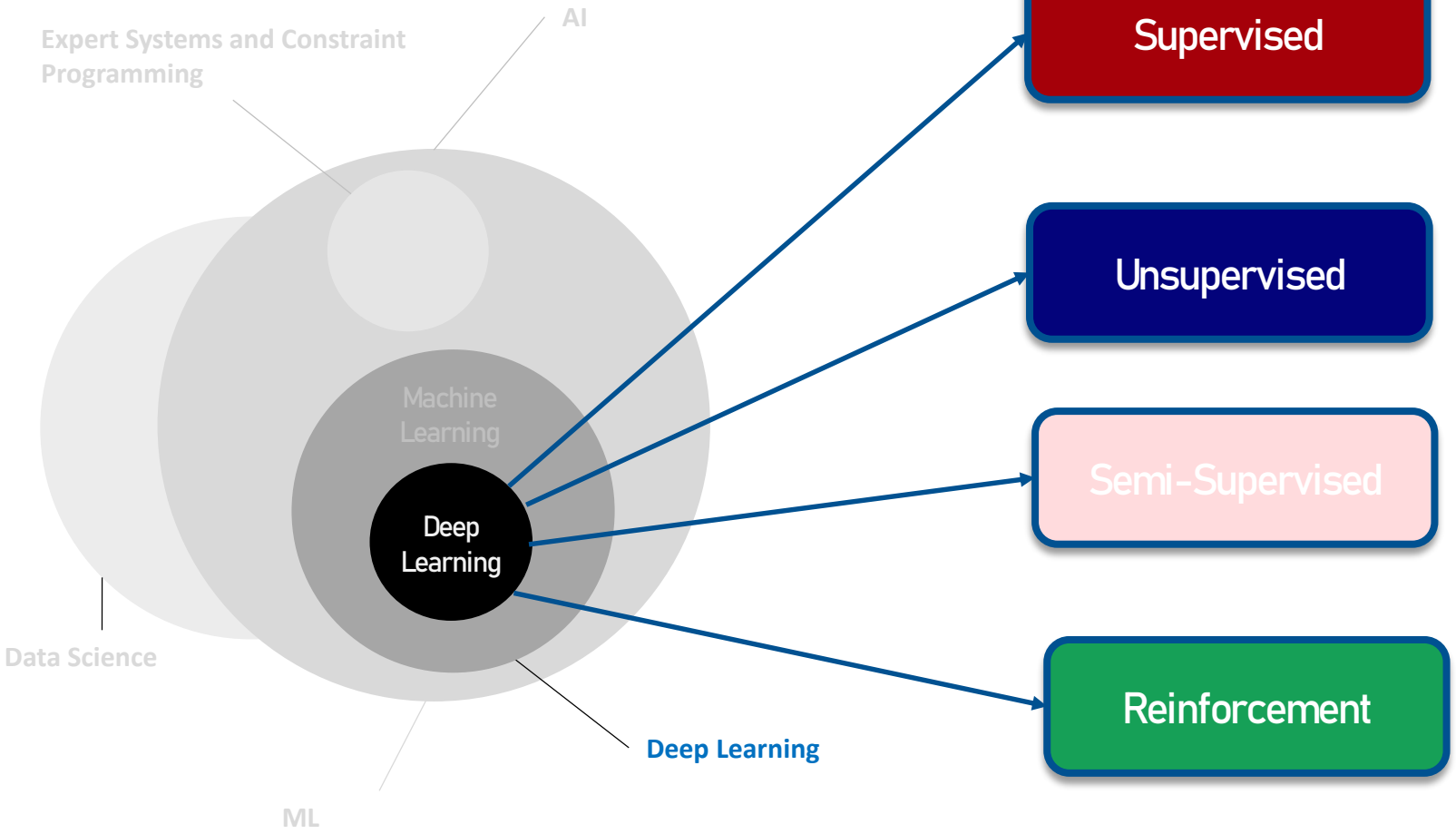
Deep Learning

=

Neural Network with hidden layers of neurons

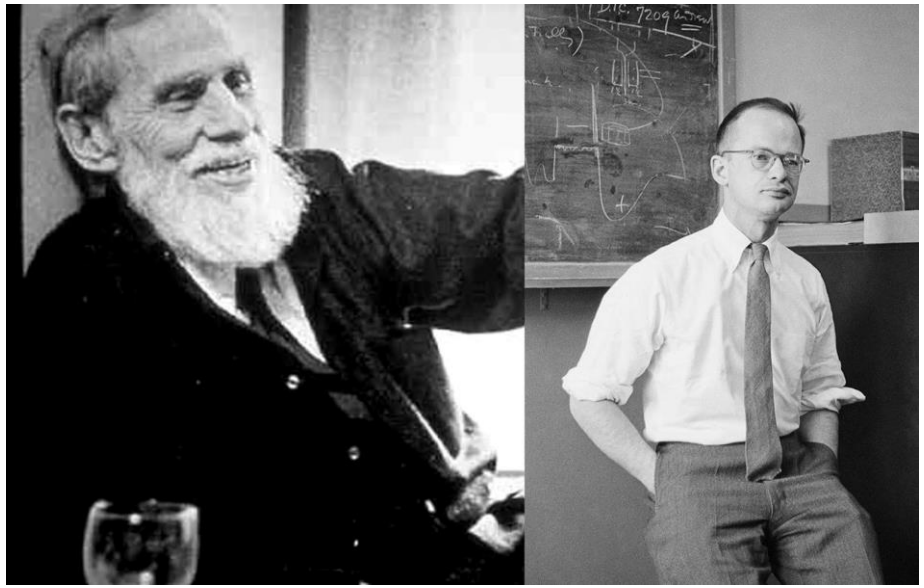
Deep learning is a subfield of machine learning that relies on models of deep neural networks. Deep learning involves training neural networks with multiple layers to learn hierarchical representations of data

Different DL models



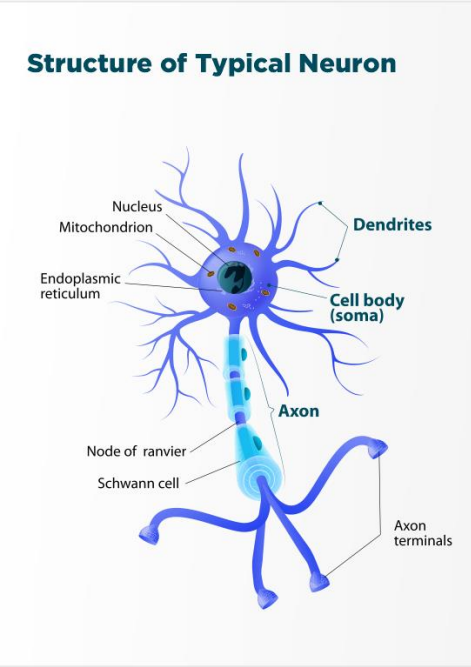
Let's go back in time

1943 - Artificial Neuron (McCulloch-Pitts)

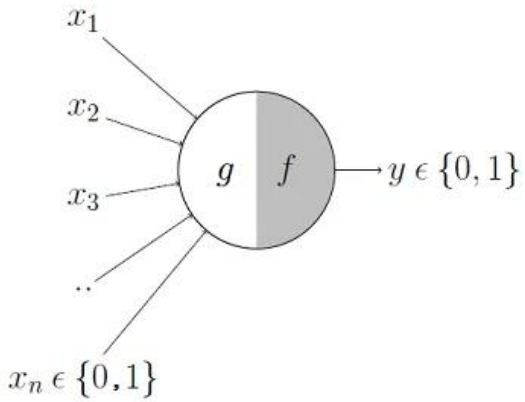


Actual neuron

Neurone M-P



Simplification

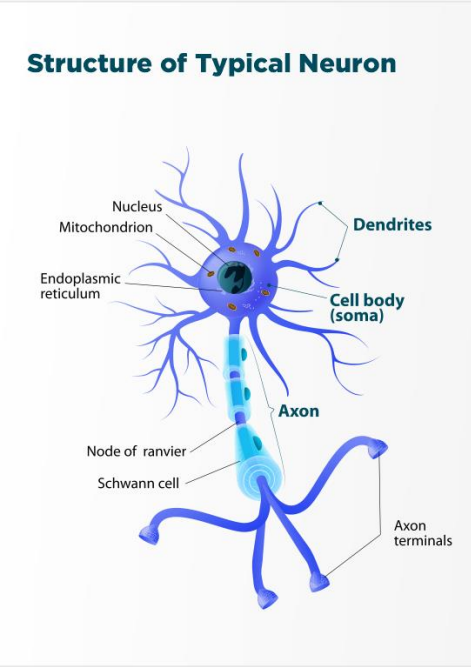


$$g(x_1, x_2, x_3, ..., x_n) = g(\mathbf{x}) = \sum_{i=1}^{n} x_i$$

$$y = f(g(\mathbf{x})) = 1 \quad if \quad g(\mathbf{x}) \geq \theta$$
$$= 0 \quad if \quad g(\mathbf{x}) < \theta$$

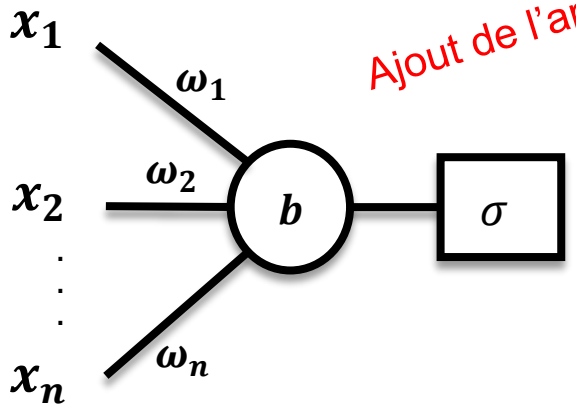
1957- Perceptron (Rosenblatt)

Neuron



Simplification

Perceptron



Ajout de l'apprentissage

McCulloch Pitts Neuron
(assuming no inhibitory inputs)

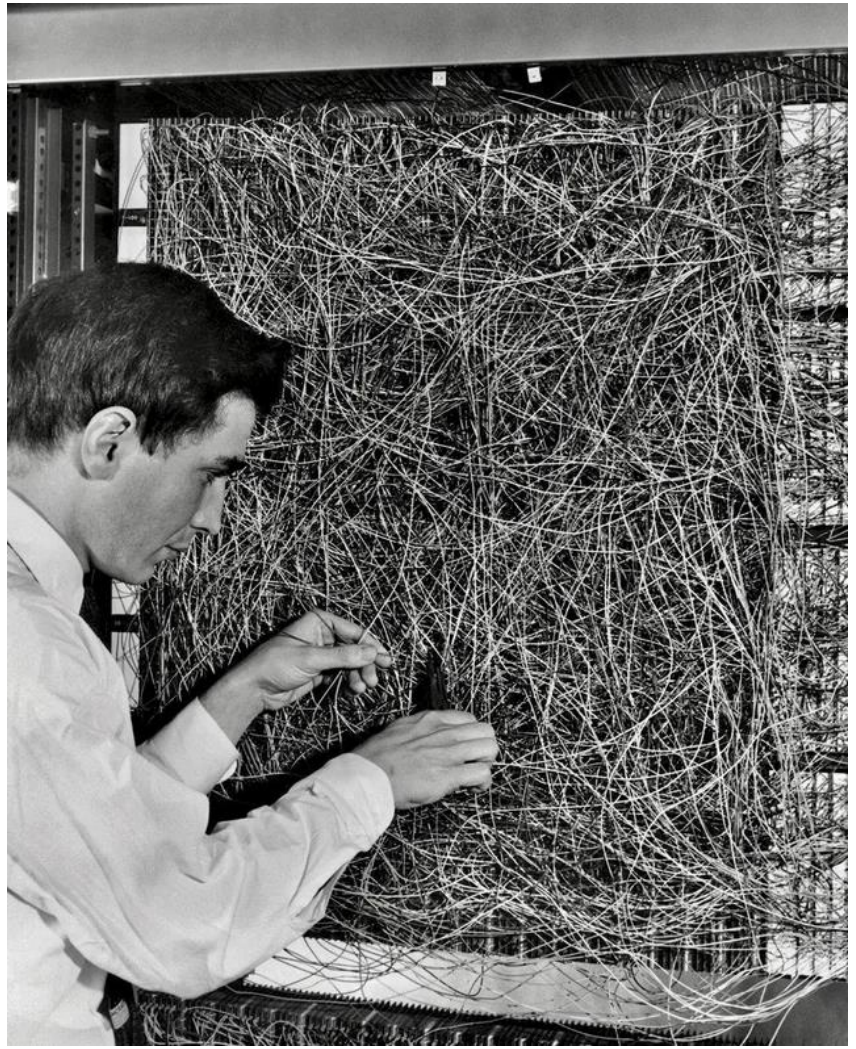
$$y = 1 \quad if \sum_{i=0}^{n} x_i \geq 0$$

$$= 0 \quad if \sum_{i=0}^{n} x_i < 0$$

Perceptron

$$y = 1 \quad if \sum_{i=0}^{n} w_i * x_i \geq 0$$

$$= 0 \quad if \sum_{i=0}^{n} w_i * x_i < 0$$

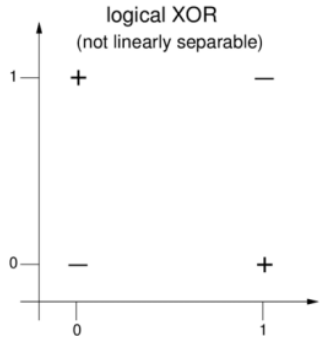


Frank Rosenblatt with a Mark I Perceptron in 1960

1969: XOR problem

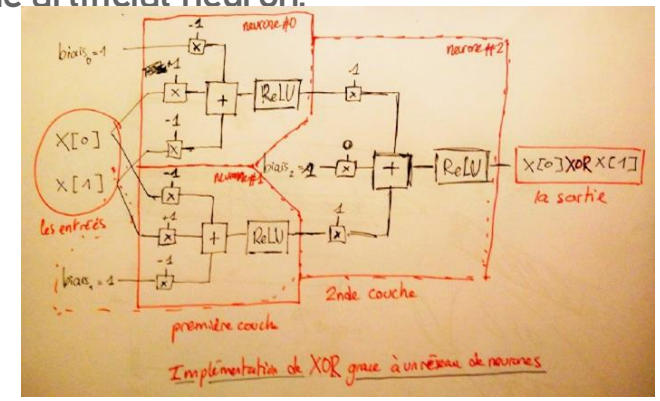
Minsky et Samuel Papert (1969)
Neurons = "mystical" character and being surrounded by a "romantic atmosphere"

- Why is XOR interesting to study?
  - Well, XOR is a problem that is much more complicated to learn than it appears because it is non-linear. It is indeed impossible to separate favorable examples from unfavorable examples with a simple straight line.
- Is it possible to solve XOR using artificial neurons ? Yes, using only two layers :
  - First layer connected to input is only two neurons
  - The second layer connected to the activations of the first layer contains only one artificial neuron.



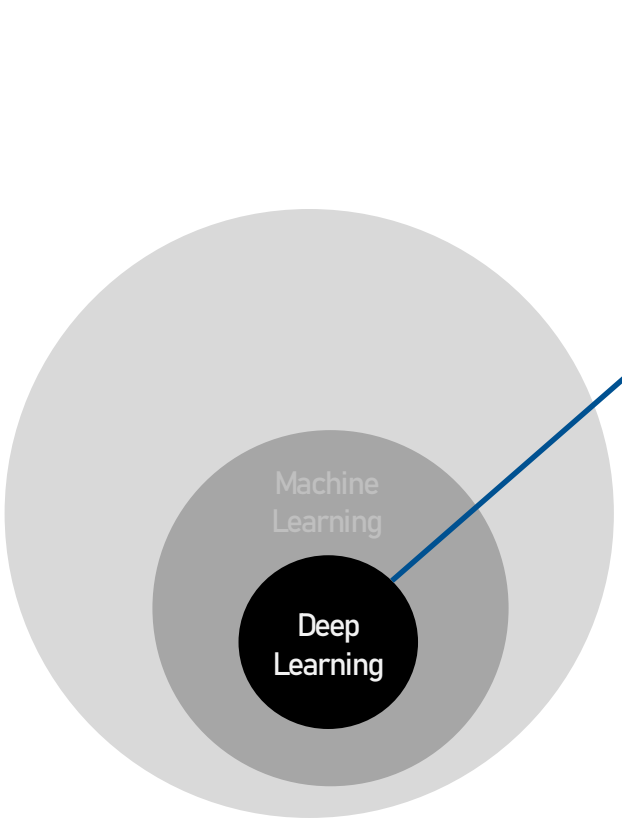
Première Couche		Deuxième Couche	
NEURONE #0		NEURONE #2	
Connexion	Poids	Connexion	Poids
X[0]	+ 1	Neurone[0]	+ 1
X[1]	- 1	Neurone[1]	+ 1
Bais	- 1	Bais	+ 0
NEURONE #1			
Connexion	Poids		
X[0]	- 1		
X[1]	+ 1		
Bais	- 1		

X[0] **XOR** X[1]



- So, we can solve nonlinear problems by stacking layers of neural networks.
- We just had a slight issue. Here, we found the solution by hand, a bit like solving a puzzle.
- To recognize rabbits in an image? It's no longer nine parameters that we have to find, but probably a few million.
- And now, it is impossible for humans to solve (unless spending a life doing so)

Different DL Models



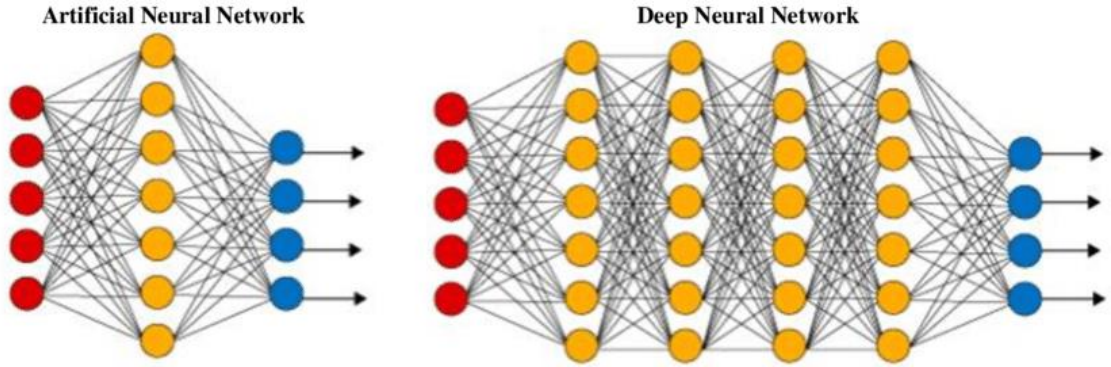
Supervised

- Deep Neural Network (DNN)
- Convolutional Neural Network (CNN)
- Reccurent Neural Network (RNN)
- Long-Short Term Memory (LSTM)

Perceptron
MLP

1980s – Deep Neural Network (DNN)

- Deep neural networks differ from classical networks (such as perceptrons) in two aspects:
  - Depth : Deep neural networks have multiple layers of artificial neurons, allowing them to learn more complex representations of data. In contrast, classical networks typically have only one or two layers.
  - Non-linearity: Deep neural networks use non-linear activation functions, such as the rectified linear unit (ReLU), which allows them to model complex relationships between inputs and outputs. Classical networks, on the other hand, use linear activation functions, which limit their ability to model non-linear relationships.



- Why this resurgence in the 2010s?
  - Growth of available computing power, and especially easy parallelization of linear algebra on special GPU (gaming) processors
  - Has made it possible to create networks with very high capacity, that is, having a very large number of parameters (on the order of several tens of millions for classical networks)

« Principe » of Deep Learning

MACHINE LEARNING

Input

Feature Extraction

Model

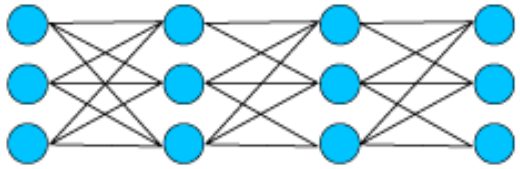
Prediction

Rabbit
Not a Rabit

Output

DEEP LEARNING

Input



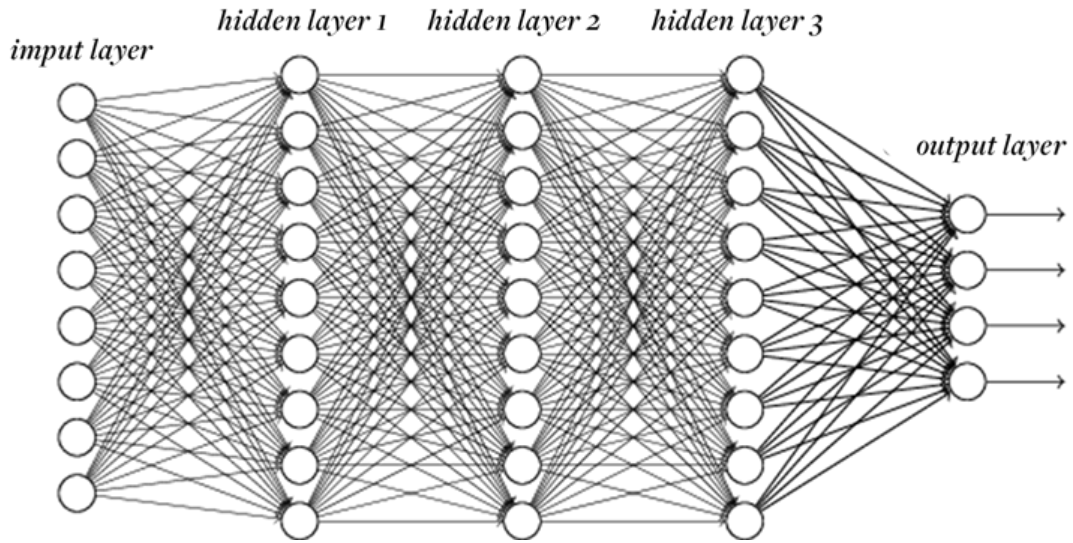
Feature Extraction + Prediction

Rabbit
Not a rabbit

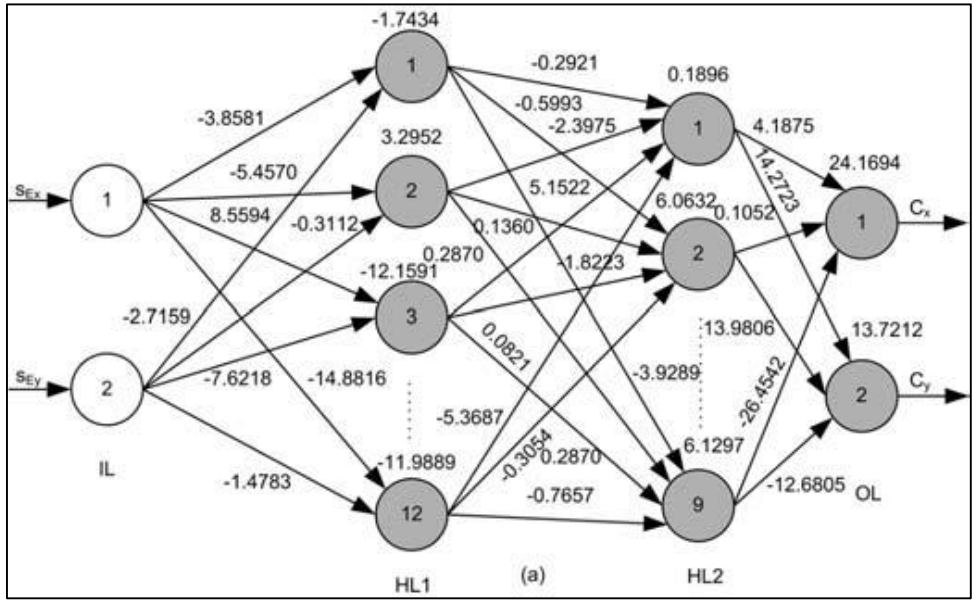
Output

1980s – Deep Neural Network (DNN)

- The operation of neural networks is very 'simplistic': the (fixed) communication between layers of artificial neurons is a weighted addition (fixed) of incoming neurons followed by an activation
- And so, those neurons are connected
- Need to define an architecture (how those neurons are connected)
- Big Net : 100 millions of connexions (parameters)

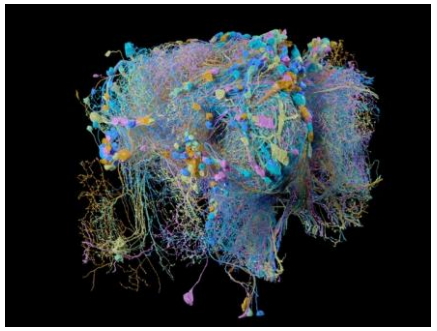


$$h(x) = f(w_i, x_i)$$

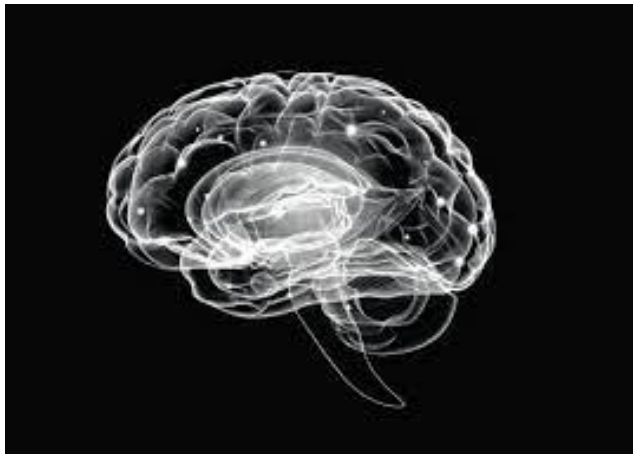


Artificial General Intelligence (AGI)

- Artificial General Intelligence (AGI) is the representation of generalized human cognitive abilities in software so that, when faced with an unfamiliar task, the AI system can find a solution
- An AGI system could theoretically perform any task that a human is capable of

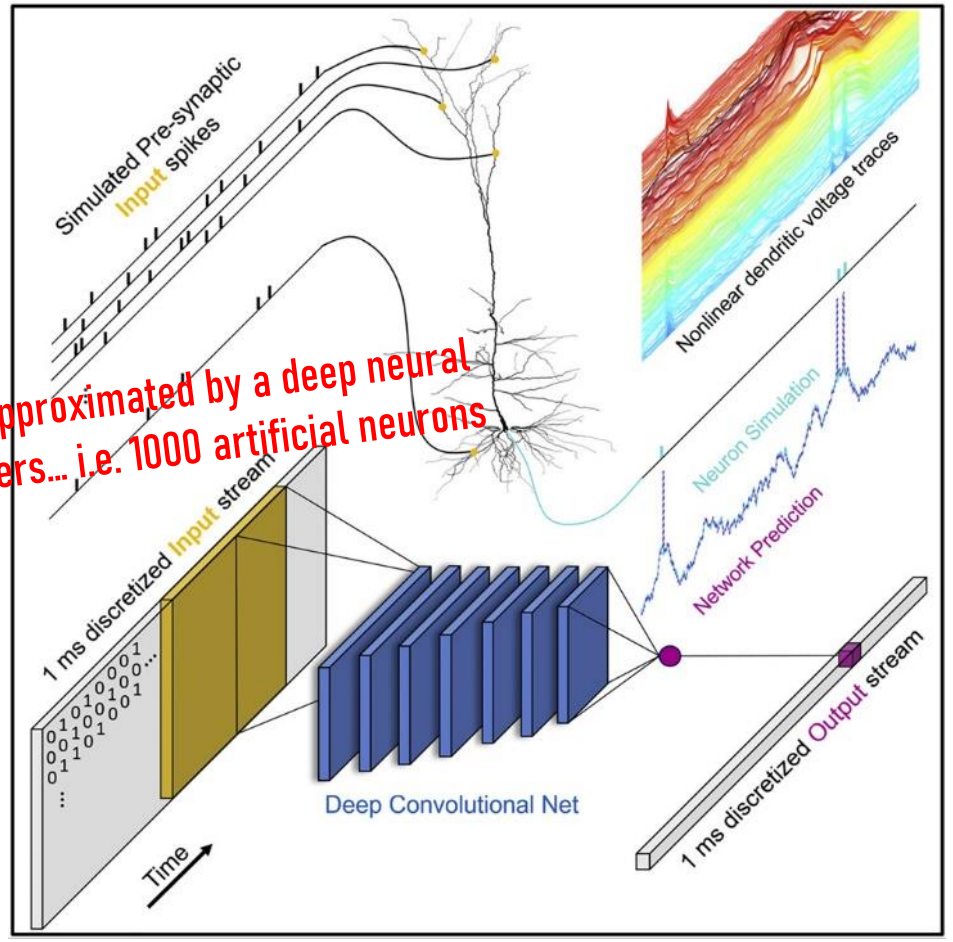


Fruits fly (613 neurons on 100 000)
12 ans with an investissement of 40 millions dollars



100 milliards of neurons
100 millions milliards connexions
0-2 ans: creation of 2 millions connexions/second

Cortical neurons are well approximated by a deep neural network (DNN) with 5-8 layers... i.e. 1000 artificial neurons

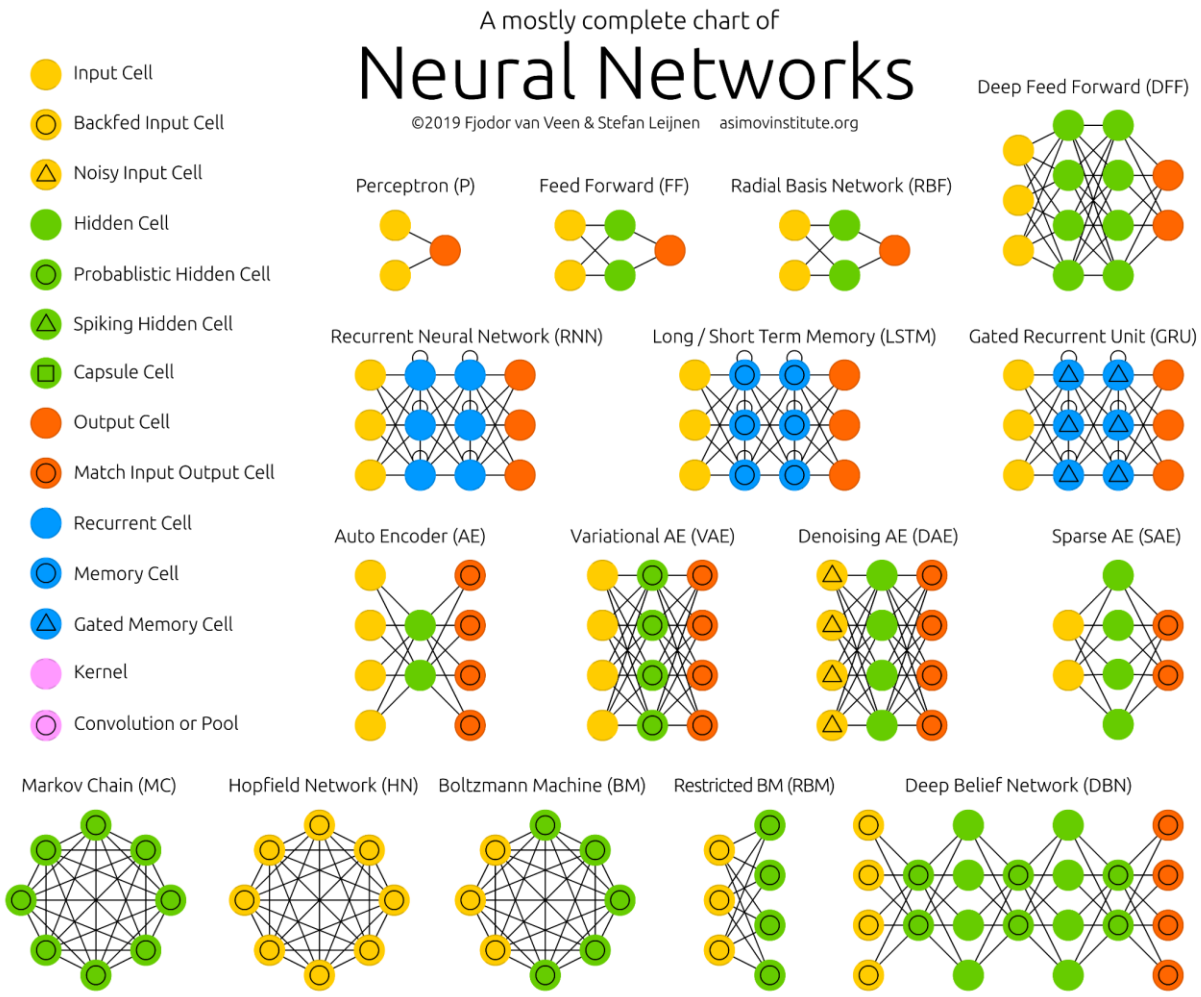


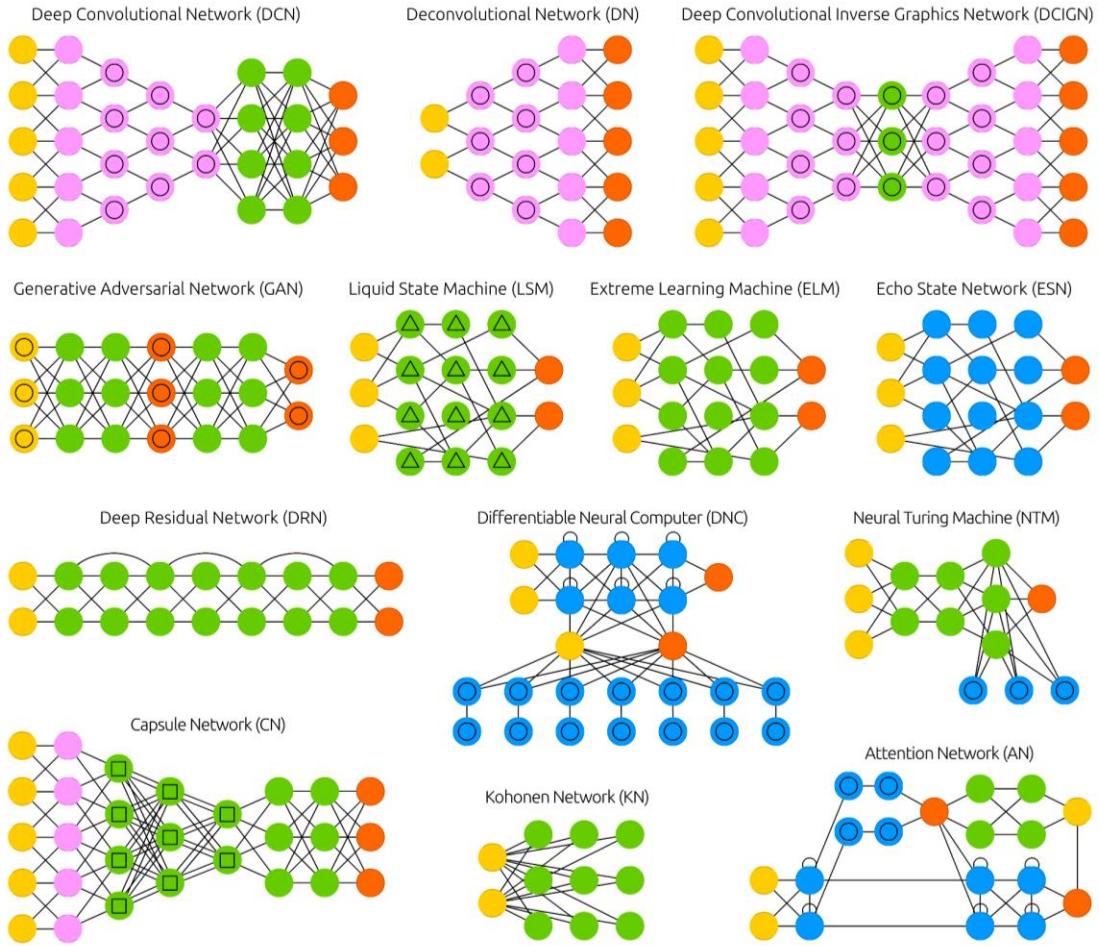
Article

Single cortical neurons as deep artificial neural networks

David Beniaguev,[1,3,*] Idan Segev,[1,2] and Michael London[1,2]
[1]Edmond and Lily Safra Center for Brain Sciences (ELSC), The Hebrew University of Jerusalem, Jerusalem 91904, Israel
[2]Department of Neurobiology, The Hebrew University of Jerusalem, Jerusalem 91904, Israel
[3]Lead contact
*Correspondence: david.beniaguev@gmail.com
https://doi.org/10.1016/j.neuron.2021.07.002

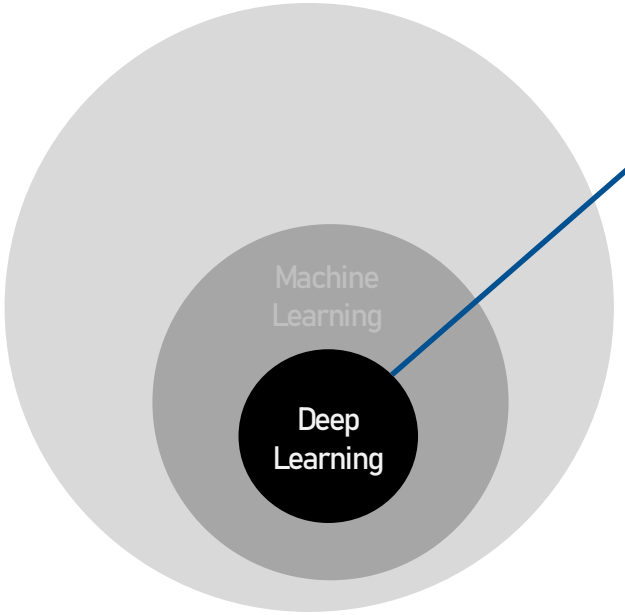
A whole bestiary !





https://www.asimovinstitute.org/neural-network-zoo/

Different DL Models



Supervised

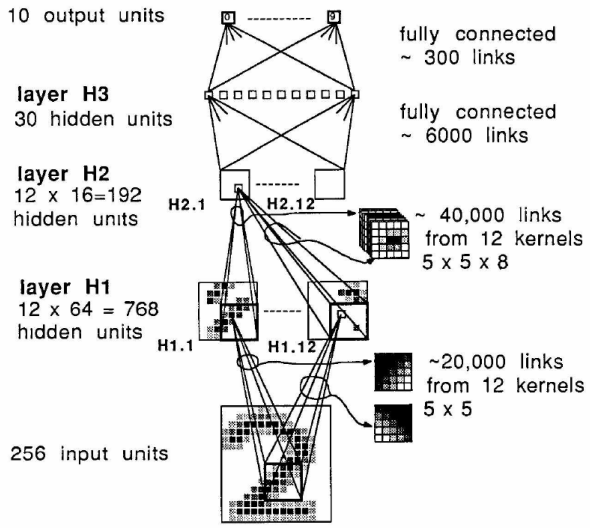
- Deep Neural Network (DNN)
- Convolutional Neural Network (CNN)
- Reccurent Neural Network (RNN)
- Long-Short Term Memory (LSTM)

Deep Learning – CNN (1989, 90s)

Backpropagation Applied to Handwritten Zip Code Recognition

Y. LeCun
B. Boser
J. S. Denker
D. Henderson
R. E. Howard
W. Hubbard
L. D. Jackel
AT&T Bell Laboratories Holmdel, NJ 07733 USA

The ability of learning networks to generalize can be greatly enhanced by providing constraints from the task domain. This paper demonstrates how such constraints can be integrated into a backpropagation network through the architecture of the network. This approach has been successfully applied to the recognition of handwritten zip code digits provided by the U.S. Postal Service. A single network learns the entire recognition operation, going from the normalized image of the character to the final classification.



And yet :

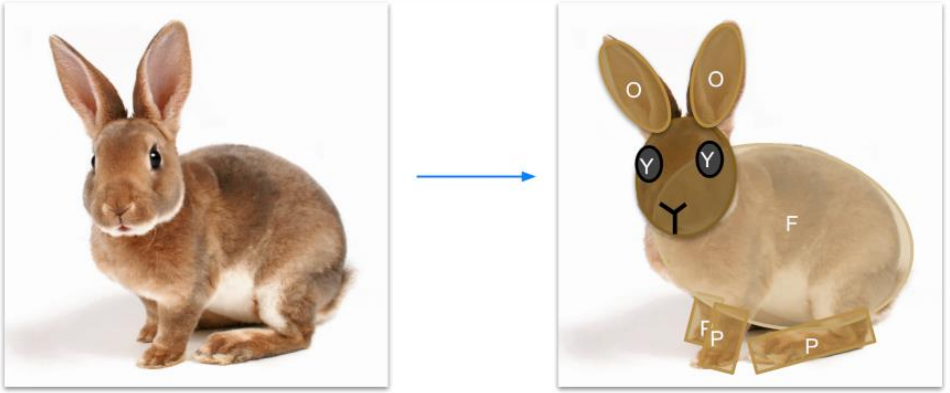
«At that time, if you said you were working on a neural network, you couldn't get a paper published. Until 2010, it was like that, a has-been thing. I remember, LeCun was a visiting professor in our lab and you had to volunteer to go eat with him. No one wanted to go. It was a curse, I swear. His papers were rejected at CVPR, his stuff wasn't in fashion, it wasn't sexy. So the guys went for the trendy stuff. They went for kernels, SVM thing. Then Lecun would say, "I have a neural network with 10 layers, it does the same thing." And we'd say, "Oh really, are you sure? What's new about it?" Because, once you've put down a neural network, okay this time it has 10 layers, but it doesn't work any better than the other one. It was crap! So he'd say, "But yes, but there's not enough data!" !"[26]. »

Interview V., chercheur en computer vision, 12 mars 2018

They always said, "Your thing is not convex, it's just a trick!" another researcher recounts. That's all they could talk about. We presented papers and they said, "It's not convex!"

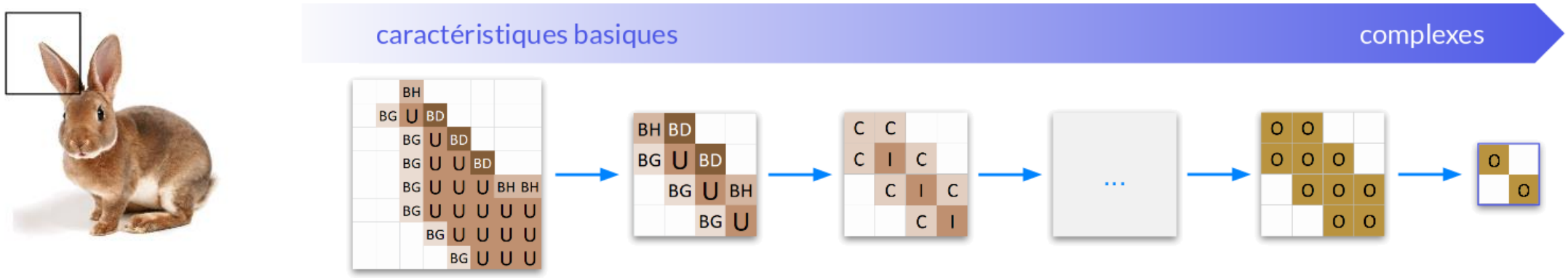
Deep Learning – CNN

- Image of 500 x 500 pixels, RGB (256,256,256)
- $(256^3)^{250\ 000}$, or $10^{1\ 806\ 225}$ different possible combination in the image!
- The number of parameters makes pixel-by-pixel analysis totally ineffective due to the complexity of the system.
- How to do it ?
- The idea is to shift to features based (or pattern) analysis
- How a human being would define a rabbit ?



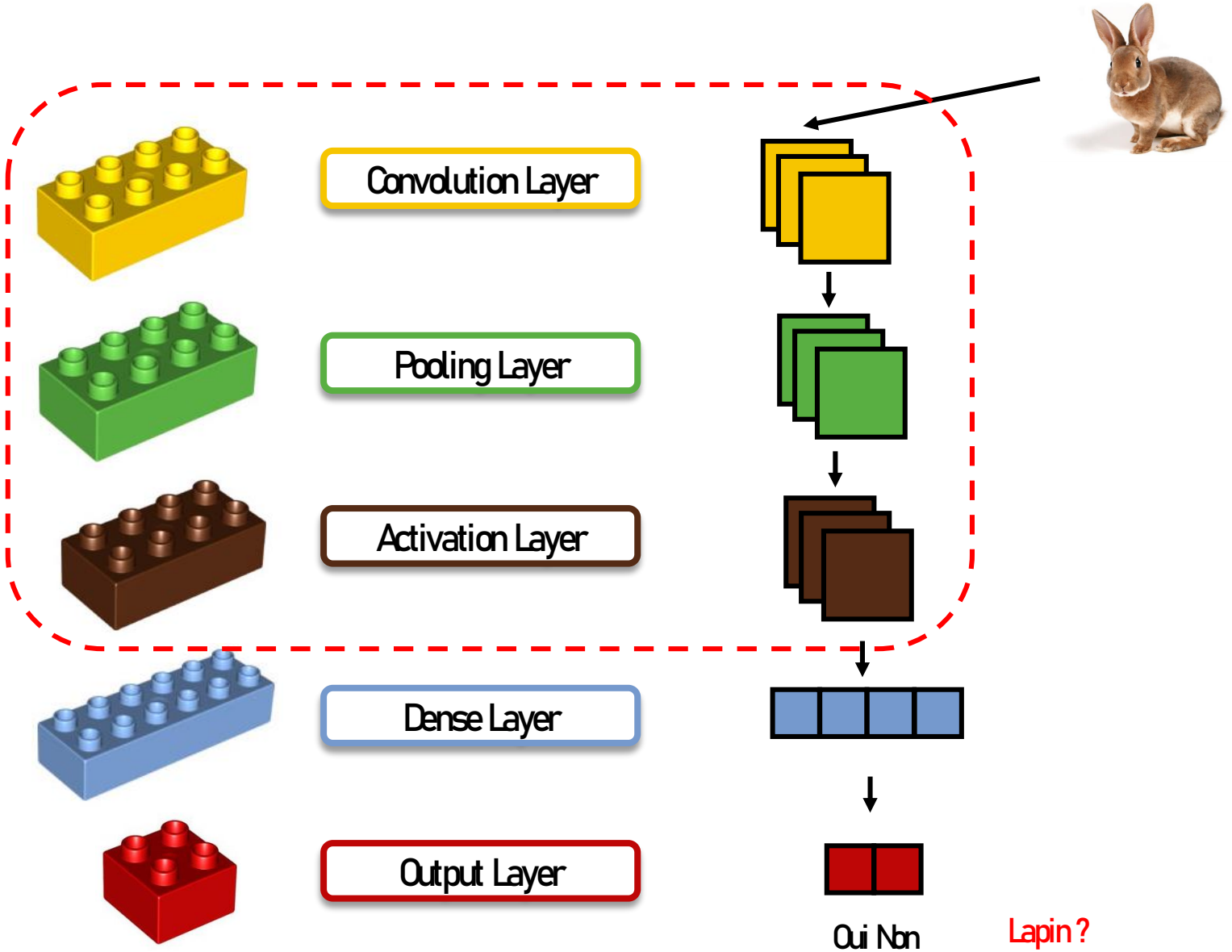
That's exactly the purpose of the convolutional neural network!
To detect in the image what makes the rabbit (with its characteristics: the two ears, the nose, the two eyes...)

Deep Learning – CNN – Features extraction



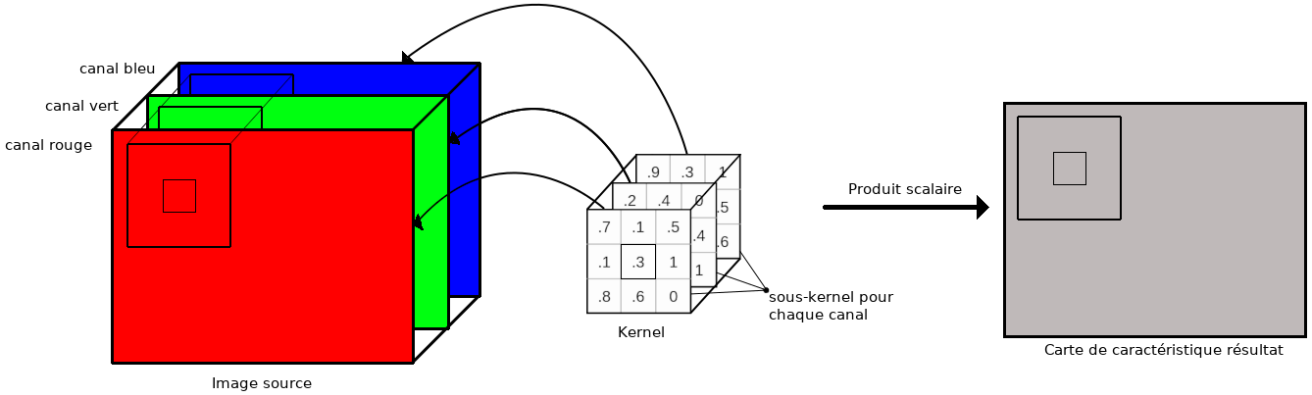
- How to translate the pixels of the rabbit's ear into the feature "ear"?
- What are the most basic characteristics that describe the image very simply?
  - Background : made of plan white
  - The edges of the ear: pixels adjacent to the border between two very different colors can be interpreted as "edges".
  - The inside of the ear : unique color of the inside
- We can correlate similar characteristics to determine new, more complex ones.
  - Ensemble of the edges are contours
  - Ensemble of united color is the inside

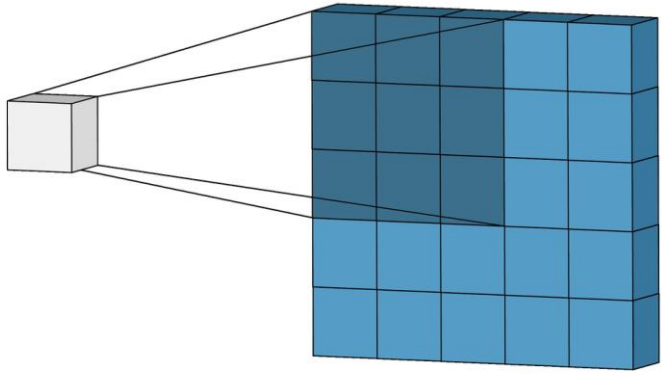
Deep Learning – CNN – Few bricks



Deep Learning – CNN – Convolutional layers

- Convolution operation consists of applying a filter to an image.
- A filter is a small matrix of different sizes (*3x3*, *5x5*, *9x9*, ...) called kernel.

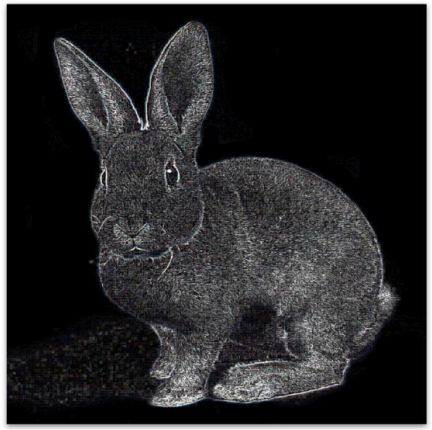




1	1	1
0	0	0
-1	-1	-1

Kernel results « front side up »

-1/8	-1/8	-1/8
-1/8	8	-1/8
-1/8	-1/8	-1/8

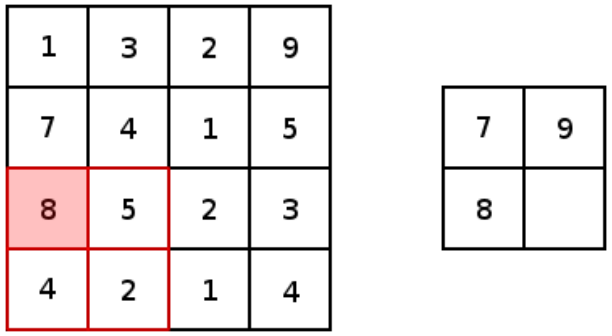


Kernel result for « accentuating details »

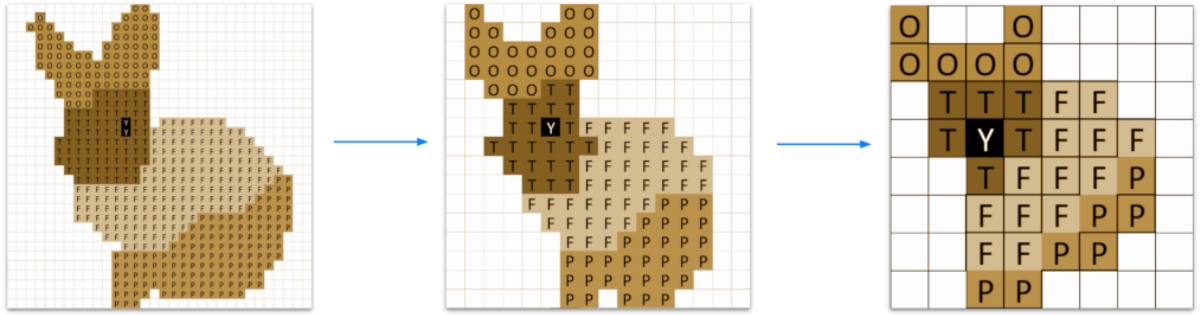
- An infinity of kernels is possible, each playing a specific and different role in highlighting a feature or improving an image
- Convolution transforms the image by highlighting certain components: the features
- The resulting image is called a feature map

Deep Learning – CNN – Pooling operation

- To make predictions on an image, the neural network does not need to know all the pixels that concern a piece of information, but rather the "ratio" of importance and its location.
- For example, a large number of pixels related to "paw" or "fur" are not necessary.
- To reduce this useless information, we use a Pooling layer
- It is a subsampling technique that involves reducing the dimension of an image while retaining the most important information.



Max Pooling example



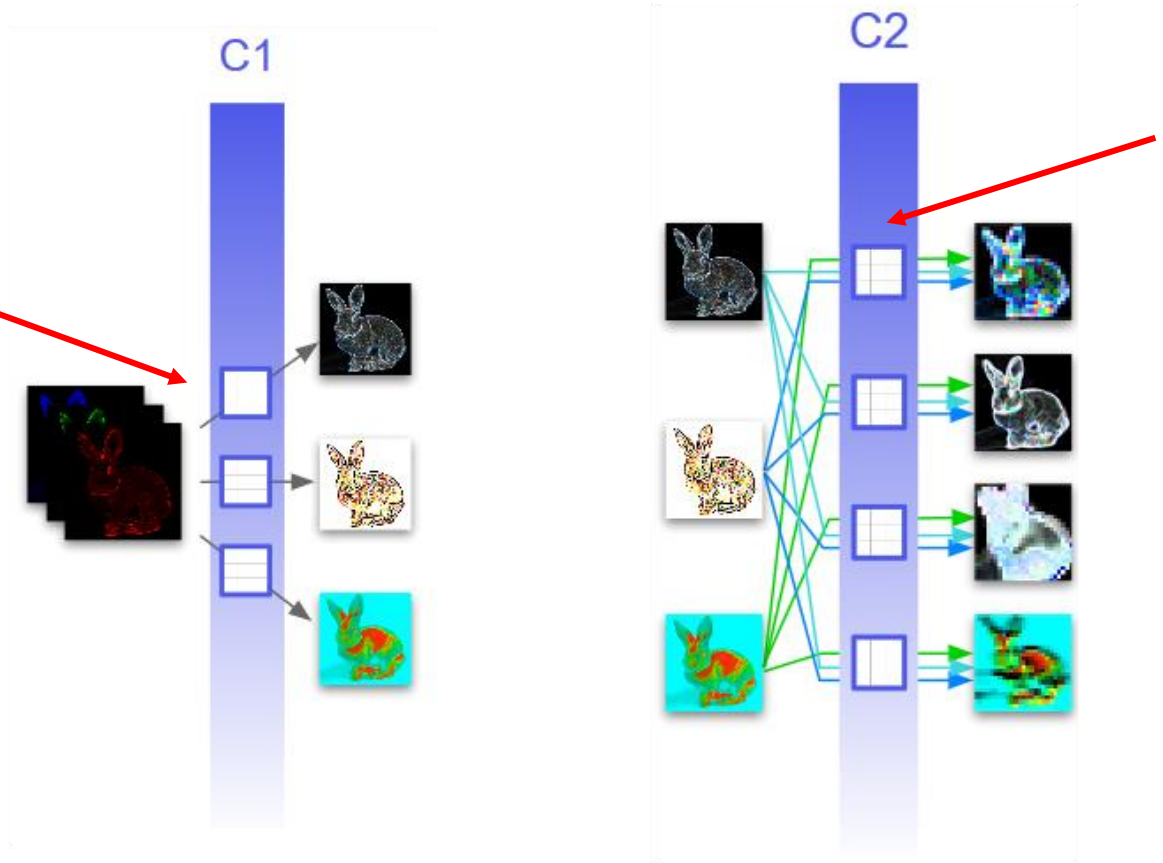
32x32 16x16 8x8

Max Pooling, Average Pooling, L2, Overlapping, Spatial Pyramid Pooling, etc...

Deep Learning – CNN – From layers to network

- We can chain convolutions to increase the complexity of the features!

- The first layer "C1" takes the image as input and applies three convolution kernels to it.
- Each kernel produces a convolution map as output, and therefore we end up with three convolution maps as output.
- Each one highlights a particular characteristic of the image.

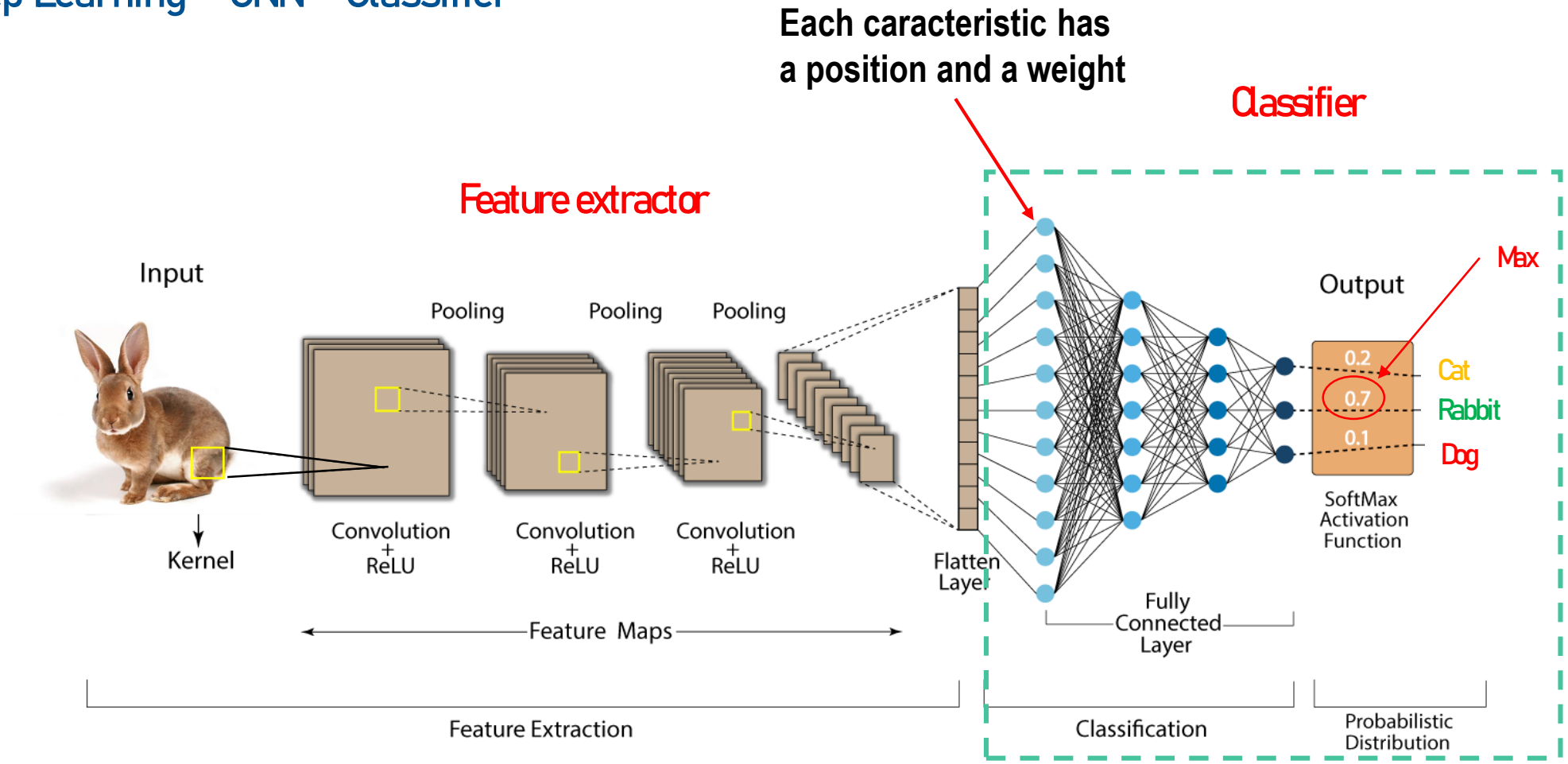


- We have three feature maps forming an image of depth 3, which we pass through four different kernels.
- Therefore, there will be 4 convolution maps as output.

And so on !

As we go further, the characteristics become more complex and gradually lose all meaning for us...

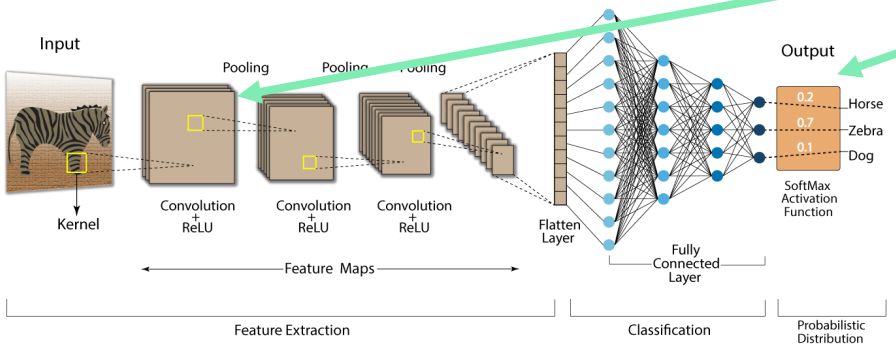
Deep Learning – CNN – Classifier



Deep Learning – CNN – Activation Map

- The activation function serves as a decision function and helps to learn complex models.
- It not only helps to learn abstractions, but also integrates non-linearity into the feature space.
- The "global function" that we are trying to approximate is logically non-linear. Therefore, we need to introduce non-linearity into the network!
- This non-linearity generates different activation patterns for different responses and thus facilitates the learning of different characteristics in images.

2 types of activations : inside the network and the final one

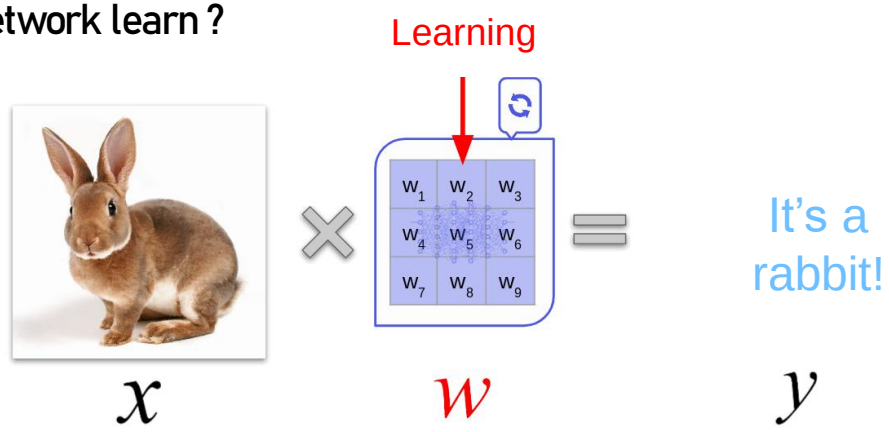


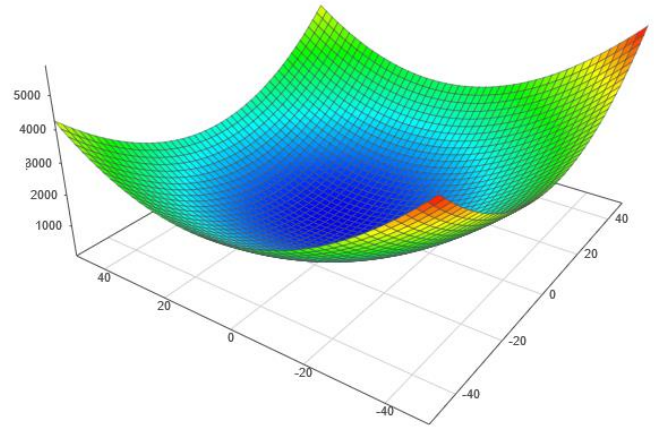
Why so many functions ?

Name	Plot	Function, $f(x)$	Derivative of f, $f'(x)$	Range	Order of continuity
Identity		x	1	$(-\infty, \infty)$	C^∞
Binary step		$\begin{cases} 0 & \text{if } x < 0 \\ 1 & \text{if } x \geq 0 \end{cases}$	$\begin{cases} 0 & \text{if } x \neq 0 \\ \text{undefined} & \text{if } x = 0 \end{cases}$	$\{0, 1\}$	C^{-1}
Logistic, sigmoid, or soft step		$\sigma(x) = \frac{1}{1+e^{-x}}$ [1]	$f(x)(1-f(x))$	$(0, 1)$	C^∞
Hyperbolic tangent (tanh)		$\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$	$1 - f(x)^2$	$(-1, 1)$	C^∞
Arctangent (arctan)		$\arctan(x)$	$\frac{1}{1+x^2}$	$\left(-\frac{\pi}{2}, \frac{\pi}{2}\right)$	C^∞
Rectified linear unit (ReLU)[9]		$\begin{cases} 0 & \text{if } x \leq 0 \\ x & \text{if } x > 0 \end{cases}$ $= \max\{0, x\} = x\mathbf{1}_{x>0}$	$\begin{cases} 0 & \text{if } x < 0 \\ 1 & \text{if } x > 0 \\ \text{undefined} & \text{if } x = 0 \end{cases}$	$[0, \infty)$	C^0
Gaussian Error Linear Unit (GELU)[4]		$\frac{1}{2}x\left(1 + \operatorname{erf}\left(\frac{x}{\sqrt{2}}\right)\right)$ $= x\Phi(x)$	$\Phi(x) + x\phi(x)$	$(-0.17\ldots, \infty)$	C^∞
Softplus[10]		$\ln(1+e^x)$	$\frac{1}{1+e^{-x}}$	$(0, \infty)$	C^∞
Exponential linear unit (ELU)[11]		$\begin{cases} \alpha(e^x - 1) & \text{if } x \leq 0 \\ x & \text{if } x > 0 \end{cases}$ with parameter α	$\begin{cases} \alpha e^x & \text{if } x < 0 \\ 1 & \text{if } x > 0 \\ 1 & \text{if } x = 0 \text{ and } \alpha = 1 \end{cases}$	$(-\alpha, \infty)$	$\begin{cases} C^1 & \text{if } \alpha = 1 \\ C^0 & \text{otherwise} \end{cases}$
Scaled exponential linear unit (SELU)[12]		$\lambda\begin{cases} \alpha(e^x - 1) & \text{if } x < 0 \\ x & \text{if } x \geq 0 \end{cases}$ with parameters $\lambda = 1.0507$ and $\alpha = 1.67326$	$\lambda\begin{cases} \alpha e^x & \text{if } x < 0 \\ 1 & \text{if } x \geq 0 \end{cases}$	$(-\lambda\alpha, \infty)$	C^0
Leaky rectified linear unit (Leaky ReLU)[13]		$\begin{cases} 0.01x & \text{if } x < 0 \\ x & \text{if } x \geq 0 \end{cases}$	$\begin{cases} 0.01 & \text{if } x < 0 \\ 1 & \text{if } x \geq 0 \end{cases}$	$(-\infty, \infty)$	C^0
Parameteric rectified linear unit (PReLU)[14]		$\begin{cases} \alpha x & \text{if } x < 0 \\ x & \text{if } x \geq 0 \end{cases}$ with parameter α	$\begin{cases} \alpha & \text{if } x < 0 \\ 1 & \text{if } x \geq 0 \end{cases}$	$(-\infty, \infty)$[2]	C^0
Sigmoid linear unit (SiLU,[4] Sigmoid shrinkage,[15] SiL,[16] or Swish-1[17])		$\frac{x}{1+e^{-x}}$	$\frac{1+e^{-x}+xe^{-x}}{(1+e^{-x})^2}$	$[-0.278\ldots, \infty)$	C^∞
Mish [18]		$x\tanh(\ln(1+e^x))$	$\frac{(e^x(4e^{2x}+e^{3x}+4(1+x)+e^x(6+4x)))}{(2+2e^x+e^{2x})^2}$	$[-0.308\ldots, \infty)$	C^∞
Gaussian		e^{-x^2}	$-2xe^{-x^2}$	$(0, 1]$	C^∞

Deep Learning – CNN – Backpropagation

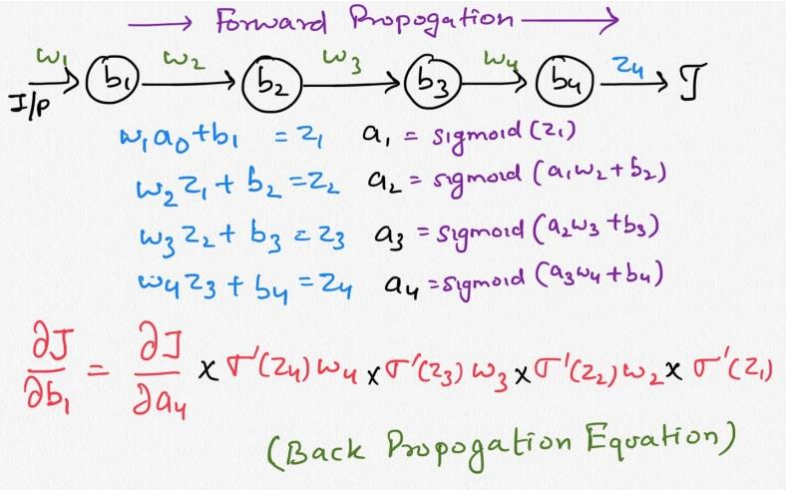
- How does the network learn ?



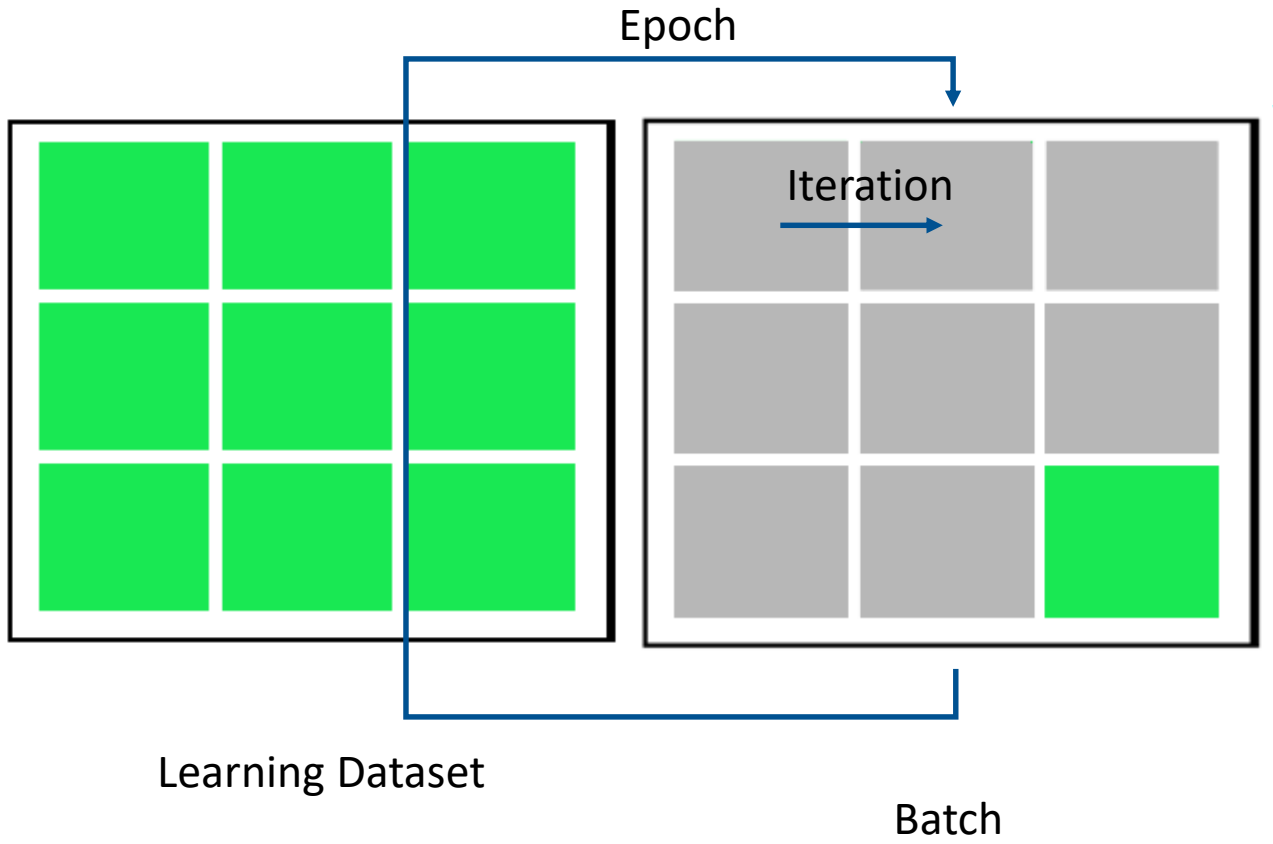


$$J(\theta_0, \theta_1) = \textbf{mean error } (h(x), y)$$

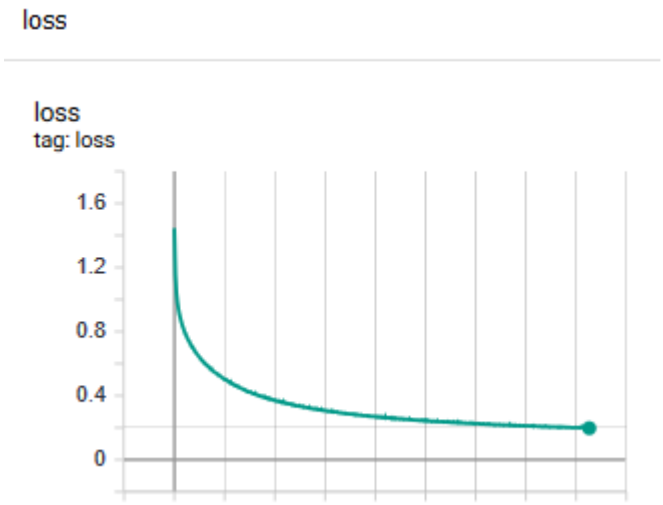
- This is where the cost function comes
- Each value of each kernel (as well as its sub-kernels) is actually a parameter of the network! Therefore, at the beginning of the network training, the kernels all have random values.
- As the network learns, each weight adapts to learn to bring out the characteristics that allow the neural network to interpret the image as accurately as possible.
- The task of learning is twofold: the network must learn to properly pre-format the image and then correctly analyze its own pre-formatting.



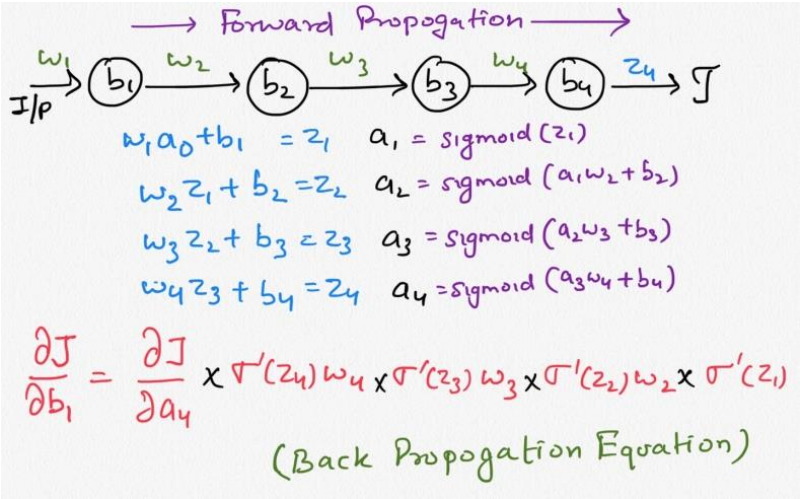
Deep Learning – CNN – Batch / Iteration / Epoch



Deep Learning – CNN – Convergence



Deep Learning – CNN – Convergence is never that easy

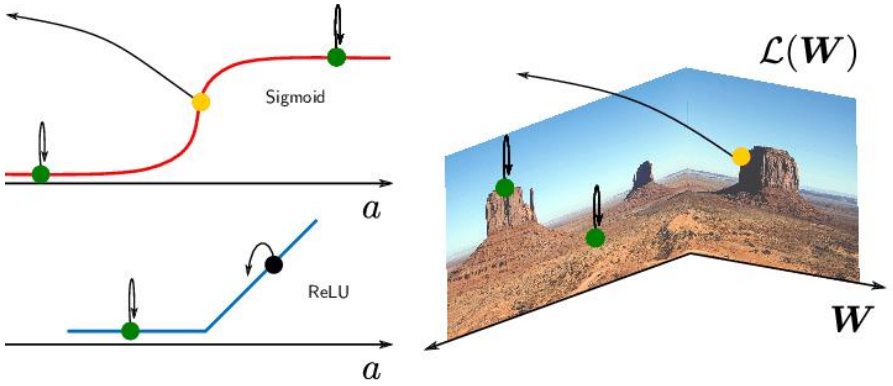


Vanishing gradient problem :

- The layers of the network are updated from the output to the input.
- As we progress towards the lower layers of the network, the gradients become smaller and smaller.
- The update by gradient descent therefore only slightly modifies the weights of the connections in the input layers, preventing good convergence of the training towards the solution.

Exploding gradient problem :

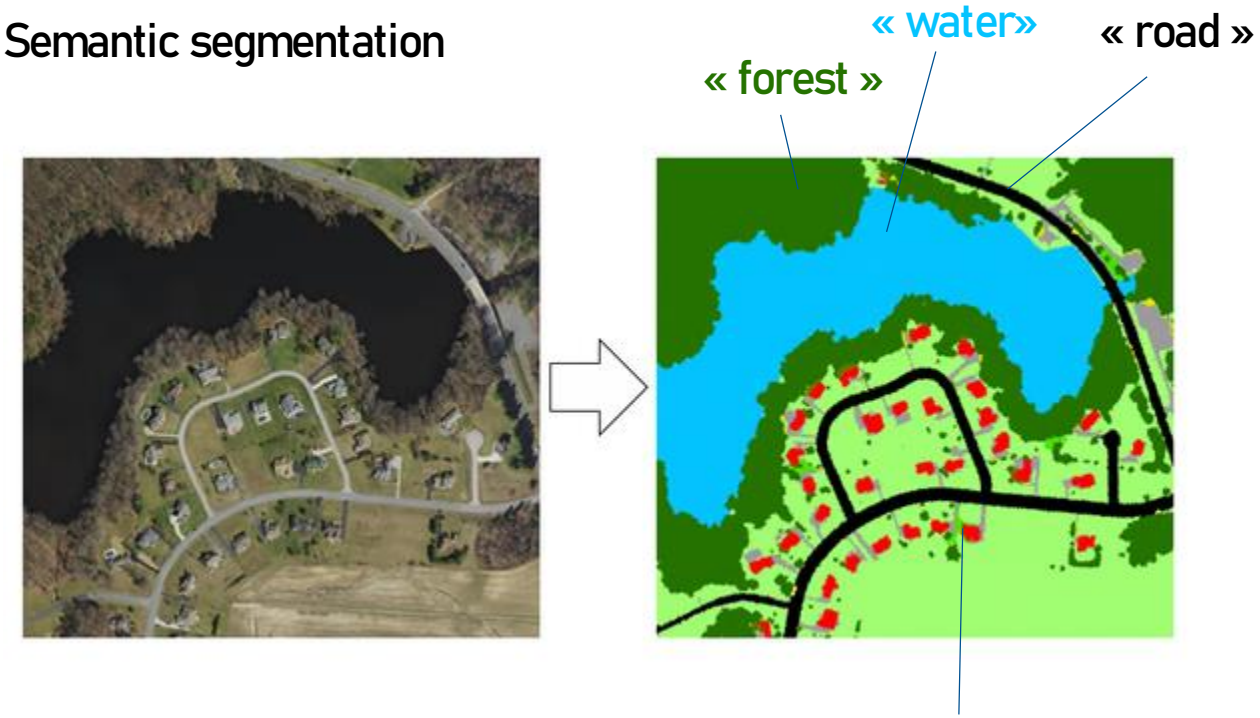
- In this case, the gradients become larger and larger.
- The layers then receive weights that are too large, causing the algorithm to diverge.



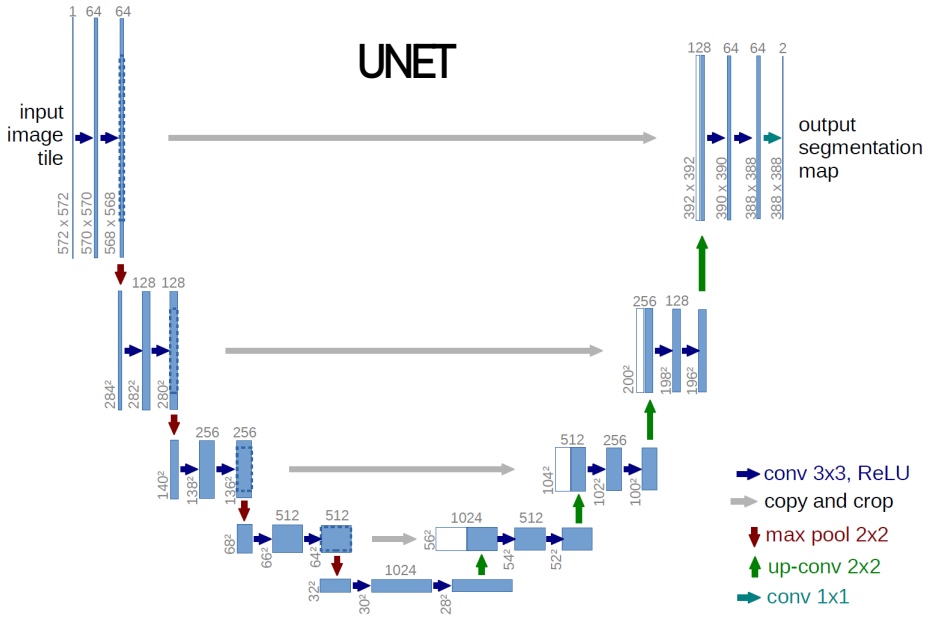
(a) Activation function (b) Loss landscape

Deep Learning – CNN – UNET

Semantic segmentation

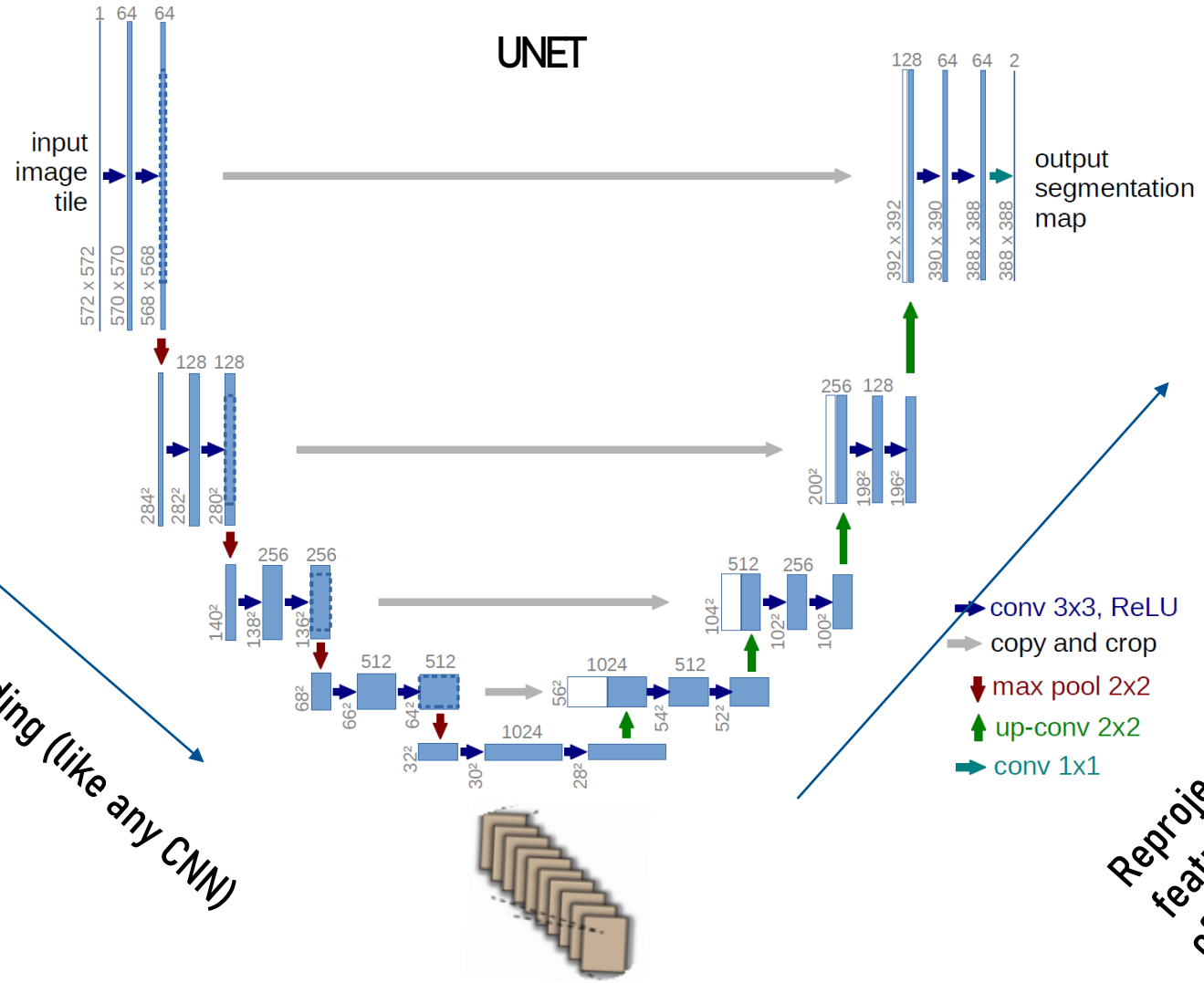


The goal of semantic segmentation is to extract relevant features from the image and use them to accurately classify the input. To achieve this, the features extracted by the network must be re-projected into a space of the same dimension as the input



Deep Learning – CNN – UNET

Semantic Segmentation



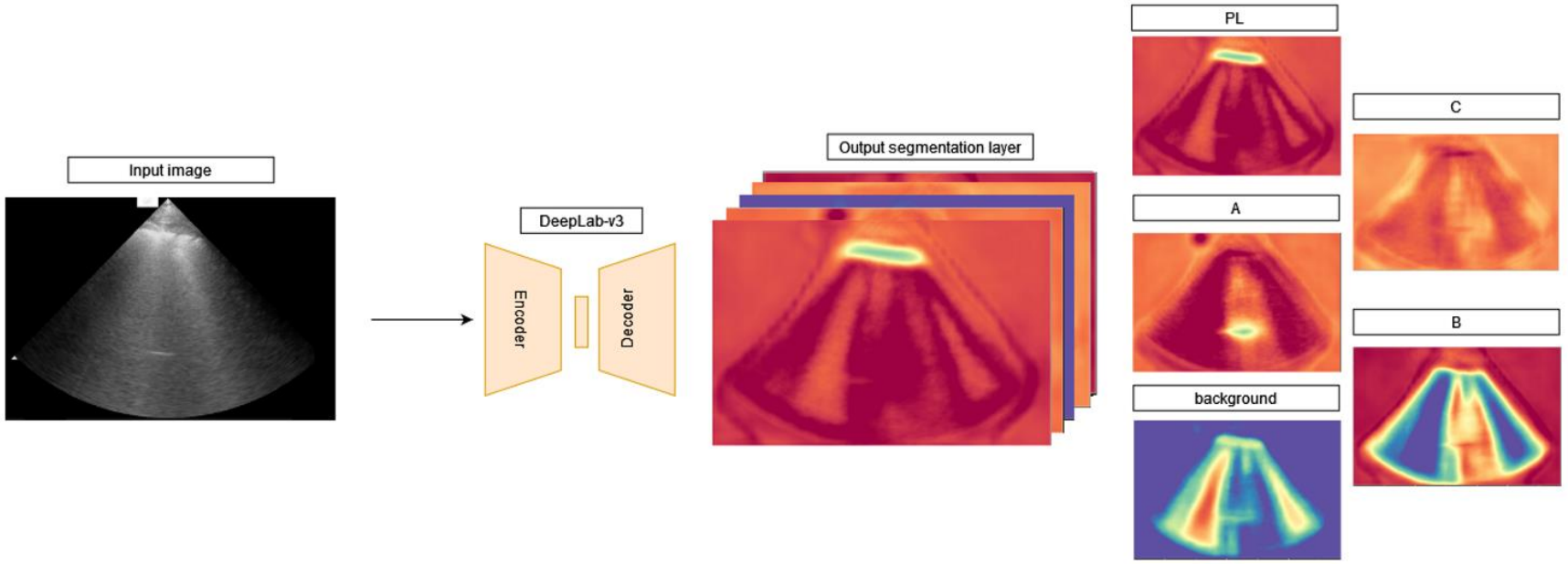
Deep Learning – CNN – CNES example

Data

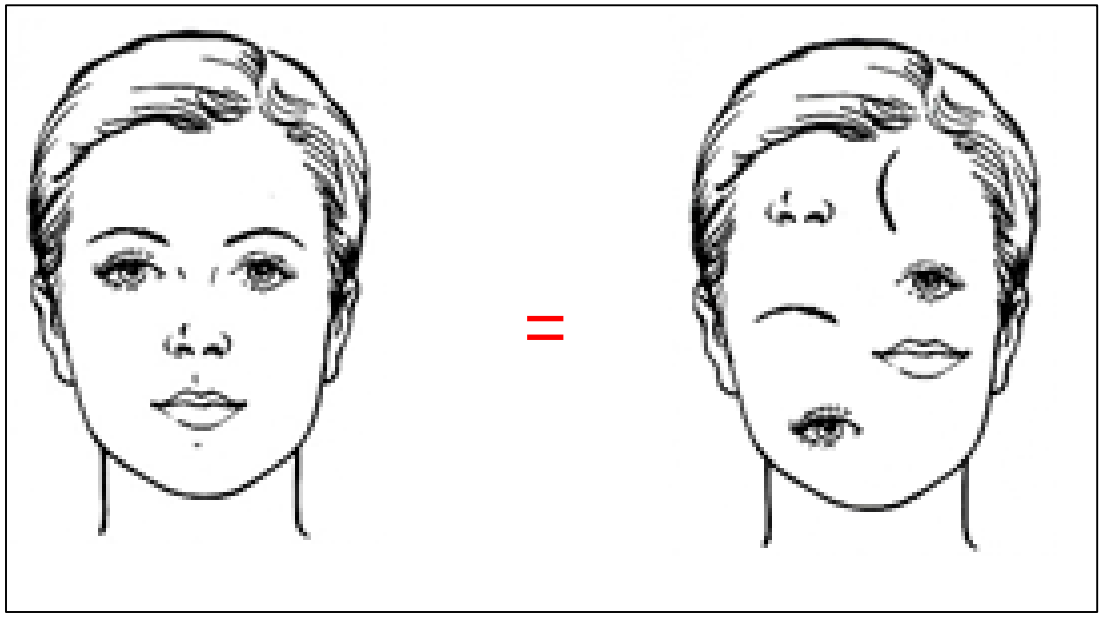
Lung Ultrasound

Objective

Quantifying the impact of a respiratory disease on the lungs using pulmonary ultrasounds (CNES - CHU).

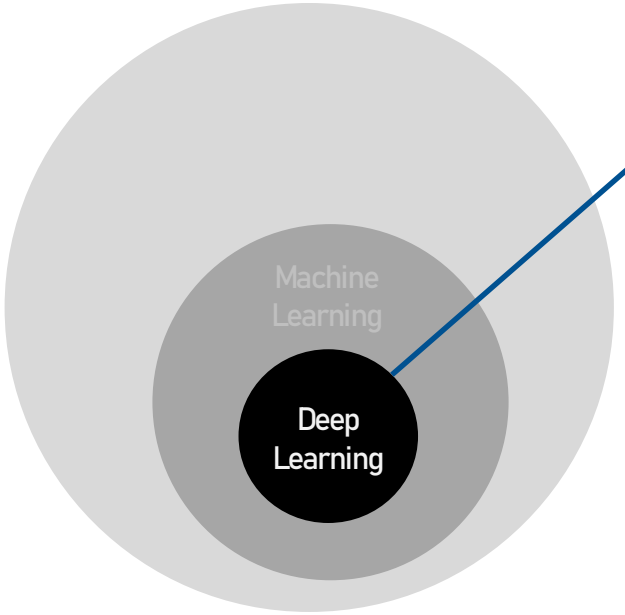


Deep Learning – CNN – Limit of CNNs



Transformers: 2020 revolution, started with NLP, then vision (ViT), more context => better results

Different DL models

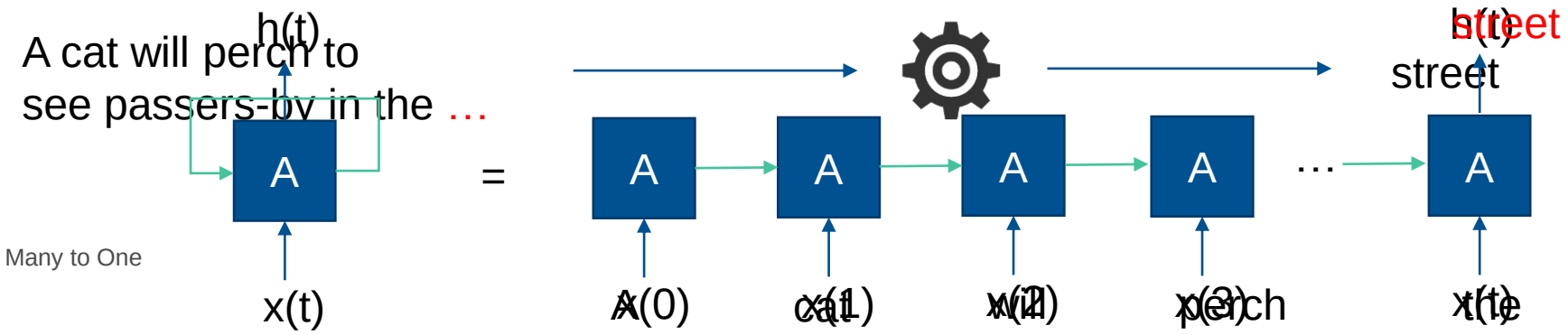


Supervised

- Deep Neural Network (DNN)
- Convolutional Neural Network (CNN)
- Reccurent Neural Network (RNN)
- Long-Short Term Memory (LSTM)

Deep Learning – Reccurent Neural Network (RNN)

❖ Mainly used to process text or time series



The text is transformed into language vectors (one word = one number)

In its naive form, RNN suffers from shortcomings:

- The backpropagation gradient becomes exponential on certain cells and hides the others
- The gradient explodes during training, preventing the network from converging

→ GRU / LSTM

Example: *« I live in Toulouse. So I speak occitan. »*

Deep Learning – Reccurent Neural Network (RNN) – CNES example

Data	USGS/S2
Objective	Answer questions on aerial/spatial data

14

Question	Ground Truth
How many commercial buildings are there?	0
What is the amount of roads?	2
Is there a commercial building at the top of a cemetery in the image?	no
How many residential buildings are there?	17
What is the amount of orchards?	0
Is there a small residential building?	yes
Are there more residential buildings than roads in the image?	yes
What is the area covered by buildings in the image?	5030m2
Is a medium building present?	yes
Are there less commercial buildings than water areas?	no

2 / 10

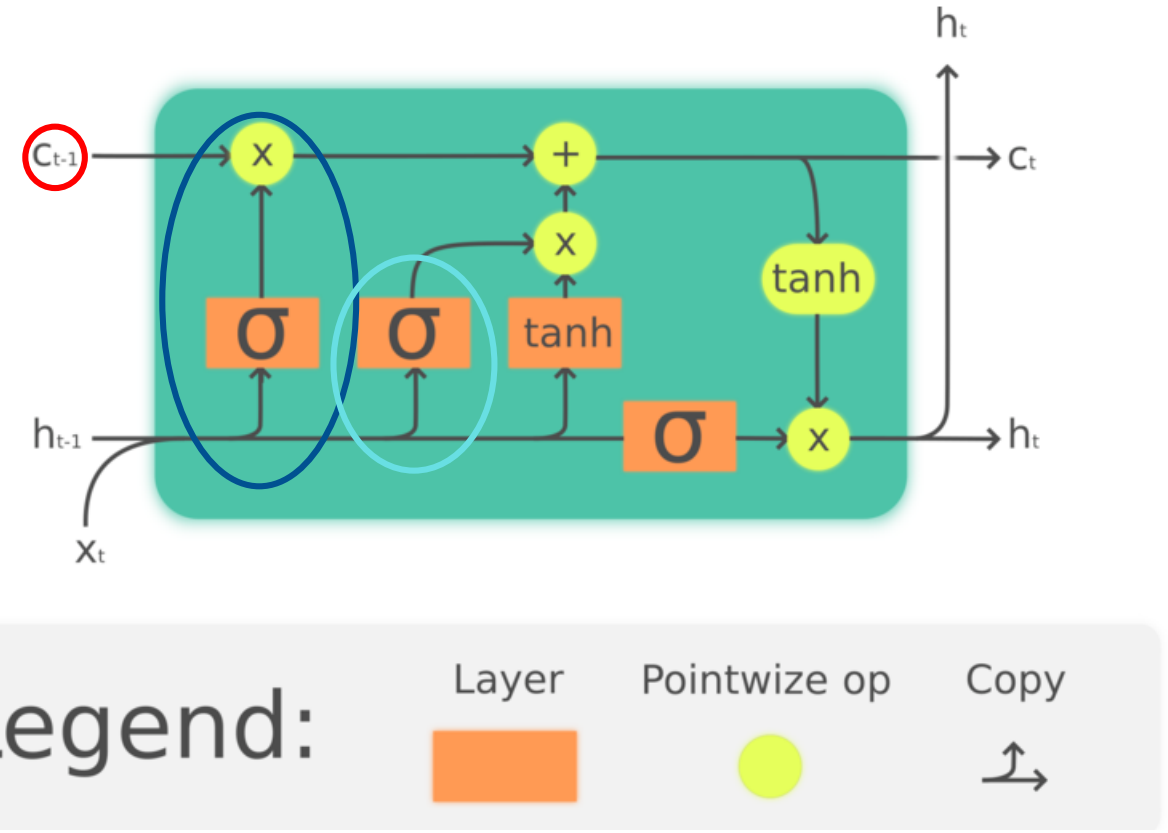
Deep Learning – Long-Short Term Memory (LSTM)

Adding better memory :

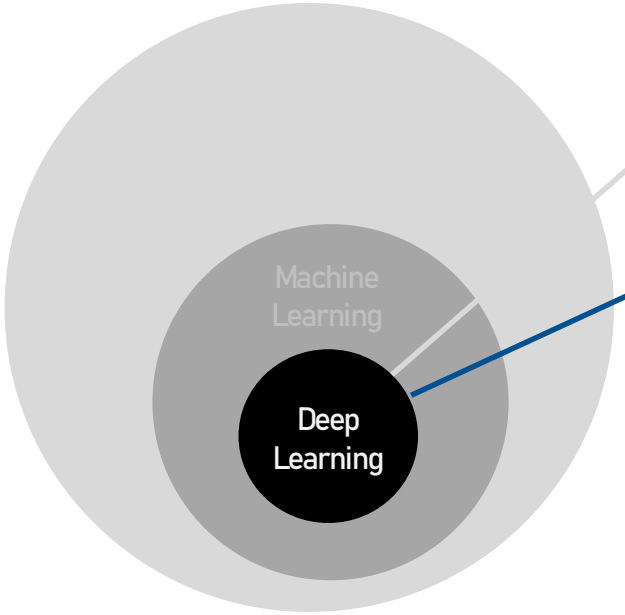
c_{t-1} is the information vector given step by step between the cells of the LSTM, it regulates the amount of information carried.

Forget gate f_t allow forgetting information that should not be communicated to c_t

Input gate i_t decides the portion of the update given by the input in each cell



Different DL models



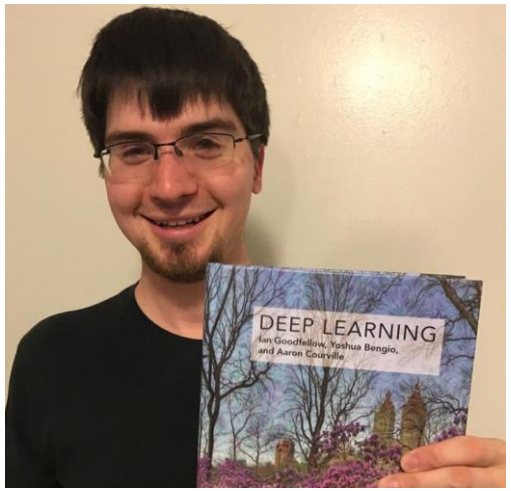
Supervised

- Deep Neural Network (DNN)
- Convolutional Neural Network (CNN)
- Reccurent Neural Network (RNN)
- Long-Short Term Memory (LSTM)

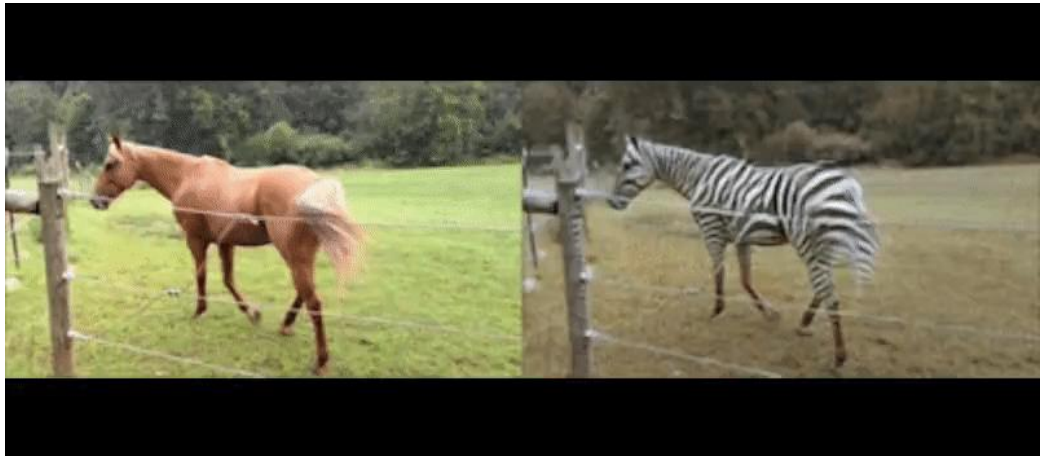
Unsupervised

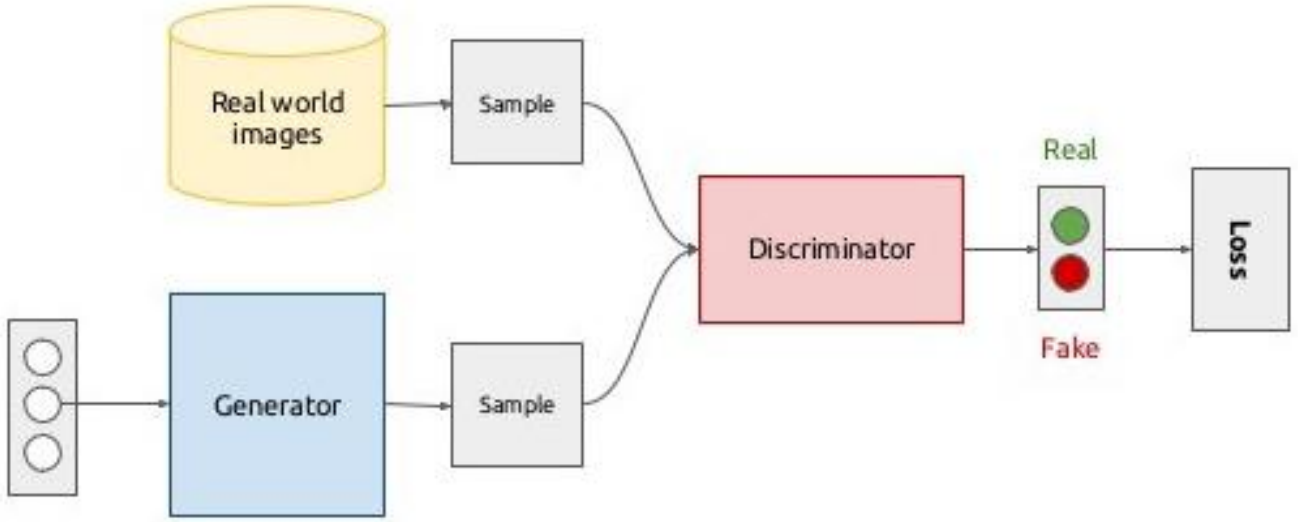
- Generative Adversarial Network (GAN)
- Reccurent Neural Network (RNN)
- Long-Short Term Memory (LSTM)
- Gated Recurrent Units (GRU)

Deep Learning – In the world of GAN...



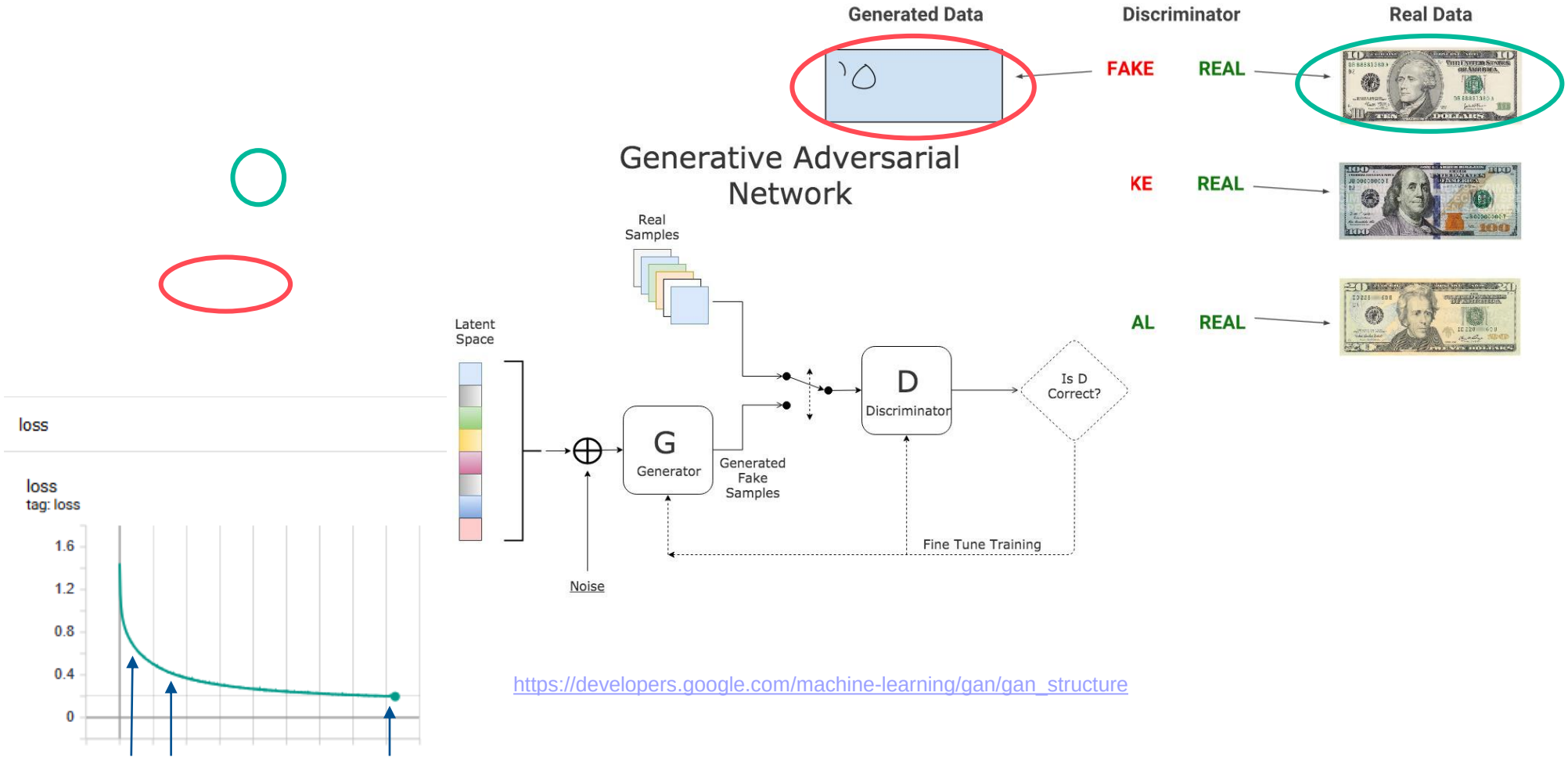
- GAN = Generative Adversarial Network
- They have been pointed out for their ability to create realistic forgeries of people, which can be used in a malicious manner by a variety of actors
- Introduced in 2014 by Ian Goodfellow, they have the particularity of being able to create data.
- GANs are composed of two networks:
  - A generator that aims to create images as realistic as possible.
  - A discriminator, a neural network that is responsible for recognizing whether the images produced by the generator are real or fake





https://thispersondoesnotexist.com/

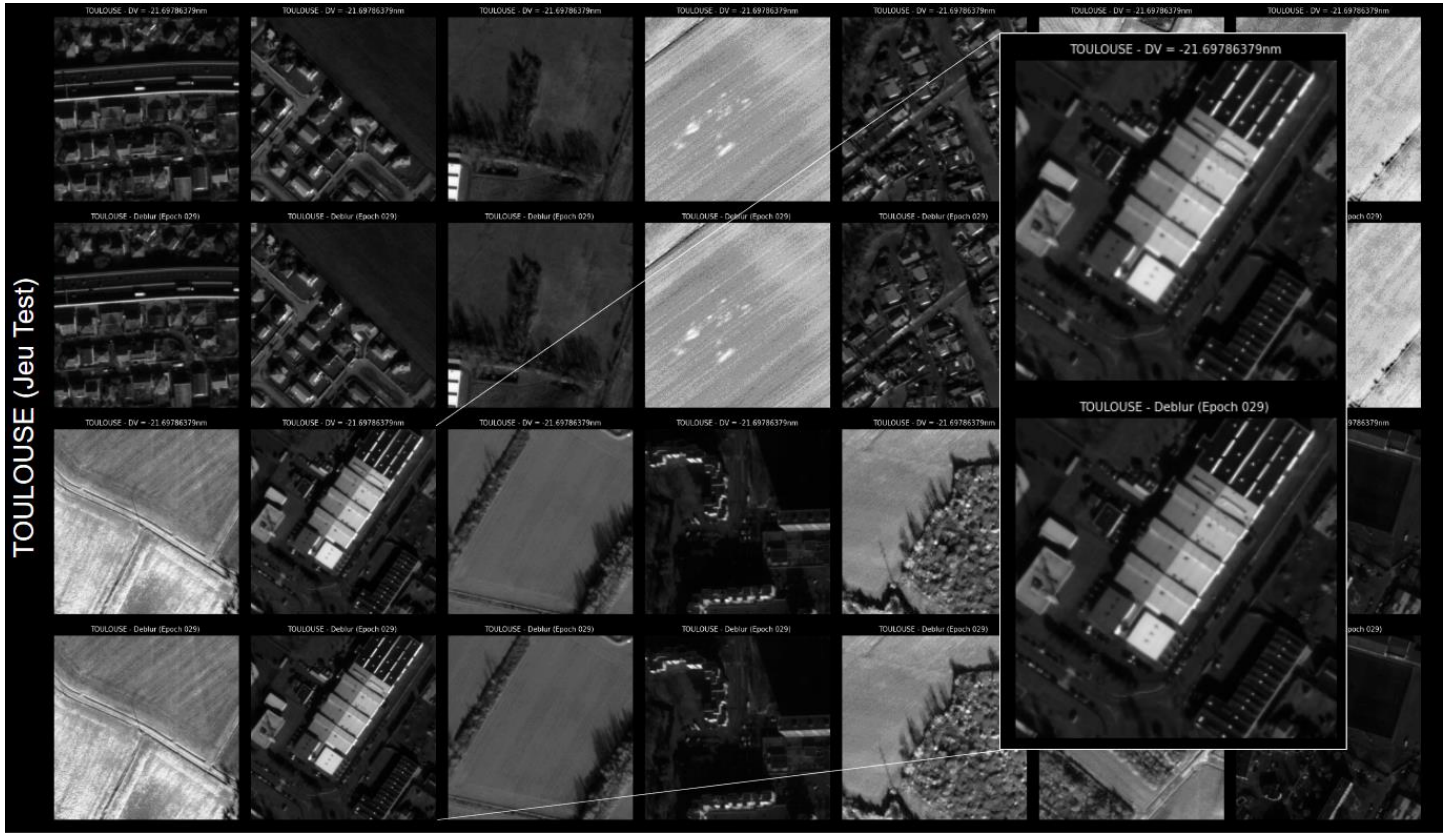
Deep Learning - GAN



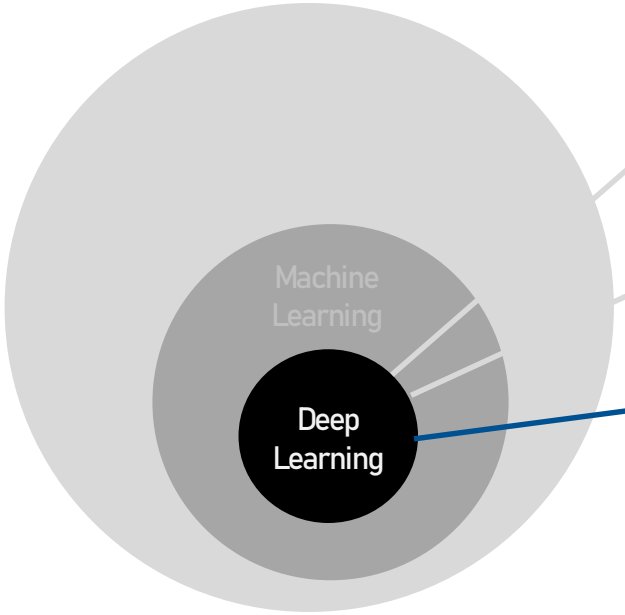
https://developers.google.com/machine-learning/gan/gan_structure

Deep Learning – GAN - CNES example

Data	Pleiades
Objective	Predicting the defocus of Pleiades optical instruments from images



Different DL models



Supervised

- Deep Neural Network (DNN)
- Convolutional Neural Network (CNN)
- Reccurent Neural Network (RNN)
- Long-Short Term Memory (LSTM)

Unsupervised

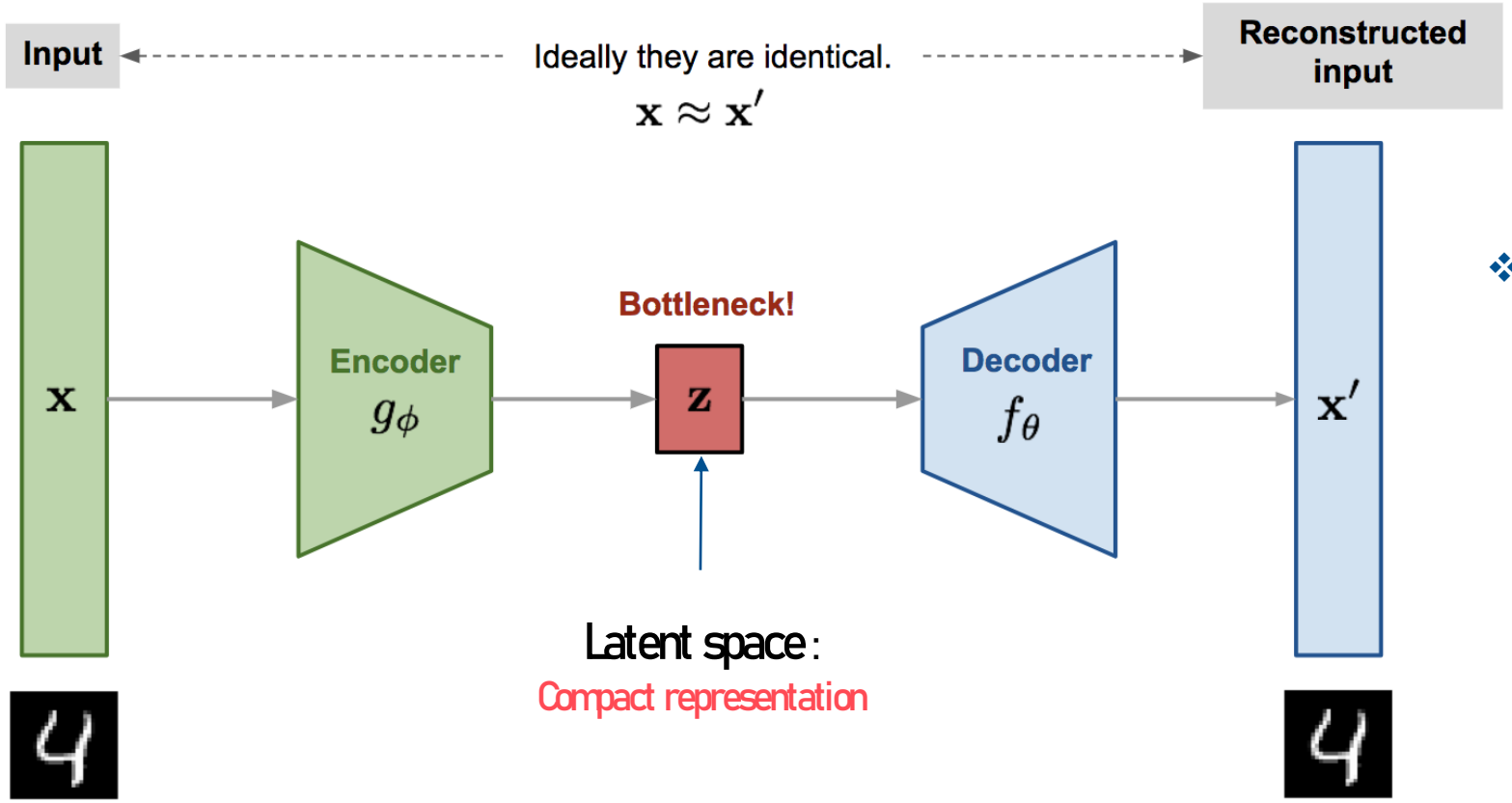
- Generative Adversarial Network (GAN)
- Recurrent Neural Network (RNN)
- Long-Short Term Memory (LSTM)
- Gated Recurrent Units (GRU)

Semi-Supervised

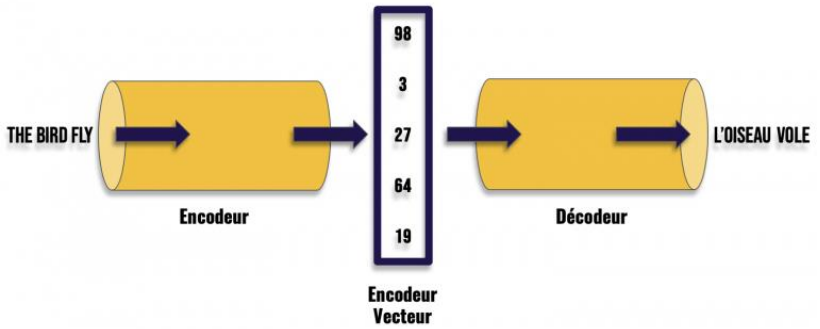
- Auto Encodeurs (AE)
- Restricted Boltzmann Machines (RBM)
- Deep Boltzmann Machines (RBM)
- Attention Network
- Transformers

Deep Learning – Autoencoder (AE)

- Autoencoders are unsupervised learning algorithms based on artificial neural networks that allow for the construction of a new representation of a dataset
- Generally, this representation is more compact and has fewer descriptors, which allows for the reduction of the dimensionality of the dataset.



- It is a fundamental pillar in translation software !

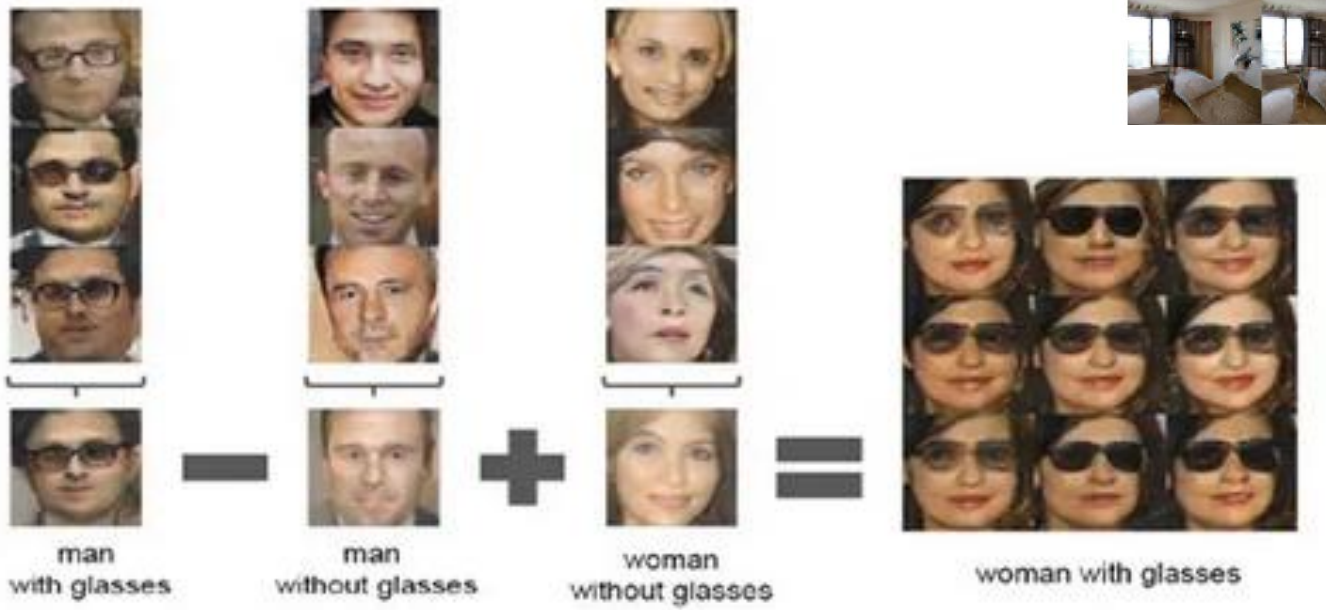


The vector created by the Encoder is fixed, which poses a problem for translating long sentences.

Deep Learning – Autoencoder (AE)

Latent Space

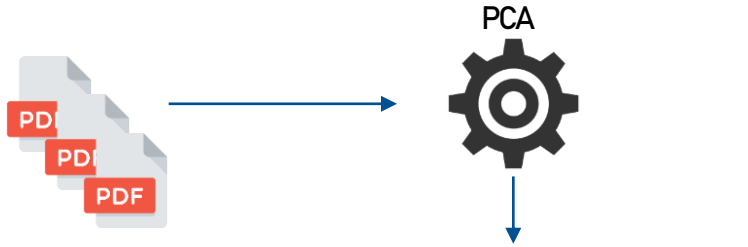
Qualitative transformation

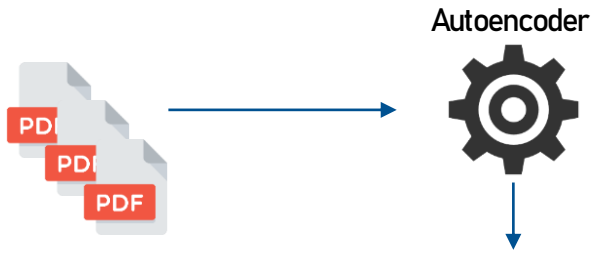


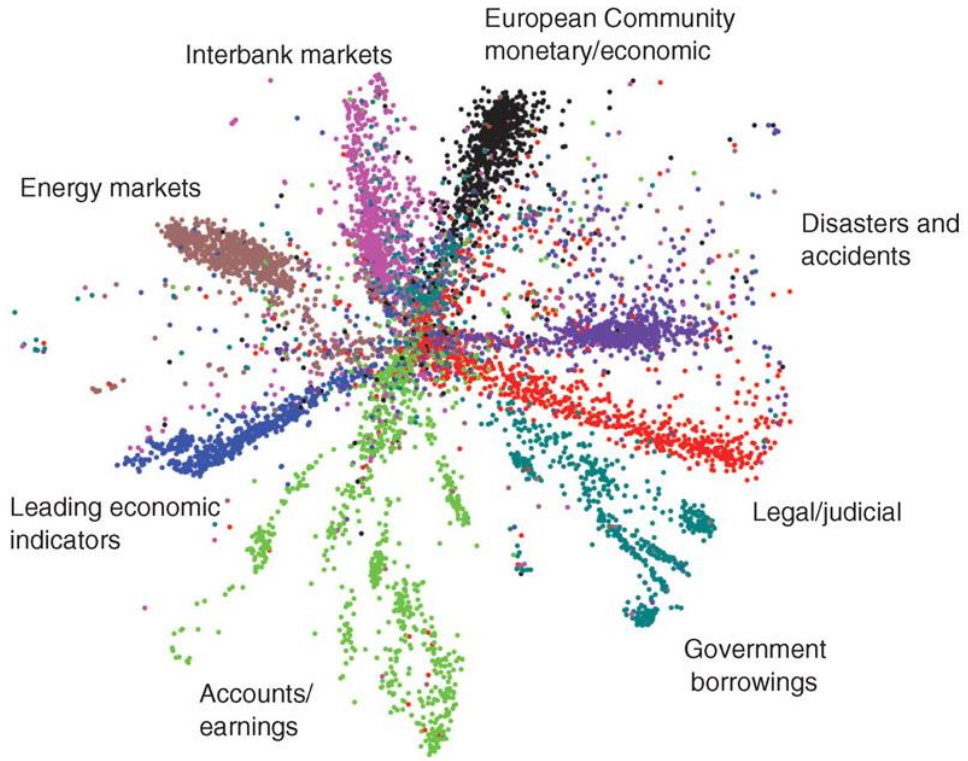
Transformation continue

"Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks", A Radford & al, 2015

Deep Learning – Autoencoder (AE) vs PCA





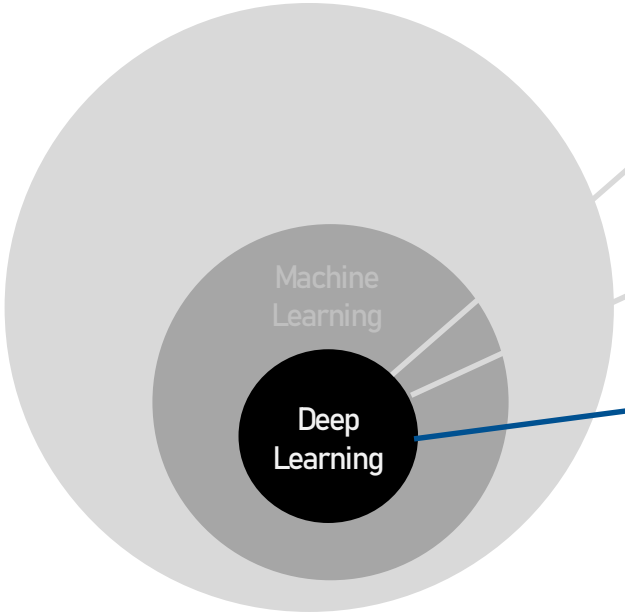


Reducing the Dimensionality of Data with Neural Networks, Geoffrey E Hinton and Ruslan Salakhutdinov, 2006

Deep Learning – Autoencoder (AE) – CNES example

Data	S2
Objective	Unsupervised classification of vine plots by grape variety

Les différents modèles de DL



Supervised

- Deep Neural Network (DNN)
- Convolutional Neural Network (CNN)
- Reccurent Neural Network (RNN)
- Long-Short Term Memory (LSTM)

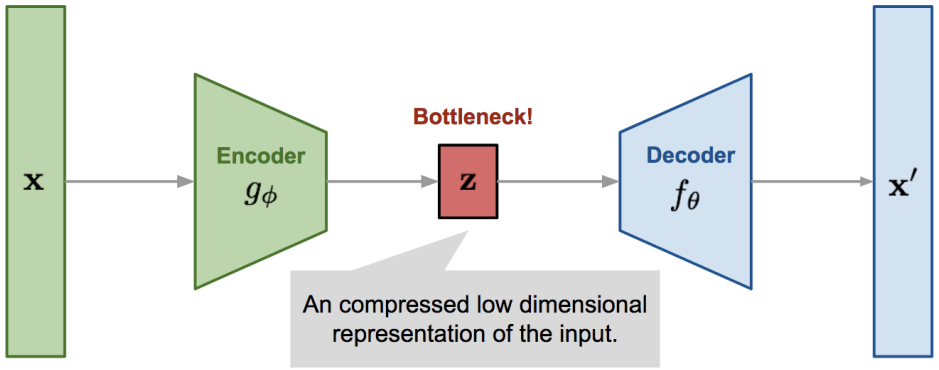
Unsupervised

- Generative Adversarial Network (GAN)
- Recurrent Neural Network (RNN)
- Long-Short Term Memory (LSTM)
- Gated Recurrent Units (GRU)

Semi-Supervisé

- Auto Encodeurs (AE)
- Restricted Boltzmann Machines (RBM)
- Deep Boltzmann Machines (RBM)
- Attention Network
- Diffusion Models
- Transformers

Deep Learning – Attention mechanism



- Inspired by human visual attention, a mechanism of attention is the ability to learn to focus on specific parts of complex data, such as a part of an image or a word in a sentence.
- Instead of focusing solely on the final output of the RNN, Attention will extract information at each step of the RNN.

NEURAL MACHINE TRANSLATION
BY JOINTLY LEARNING TO ALIGN AND TRANSLATE

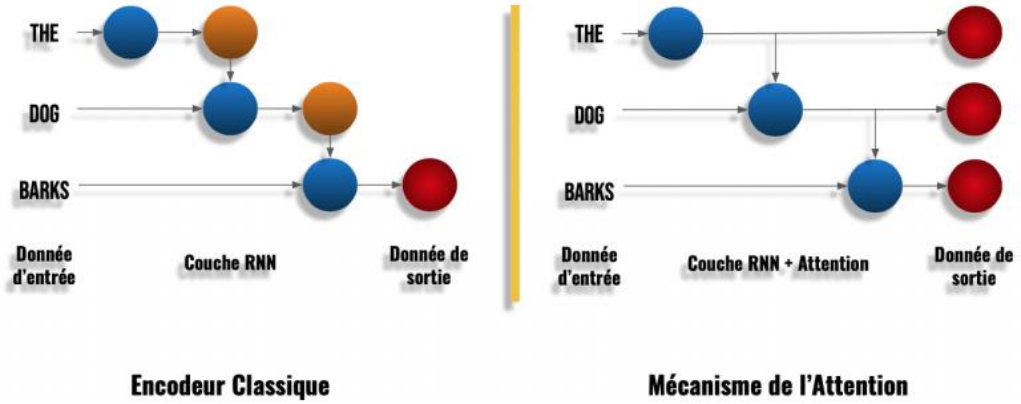
2014

Dzmitry Bahdanau
Jacobs University Bremen, Germany

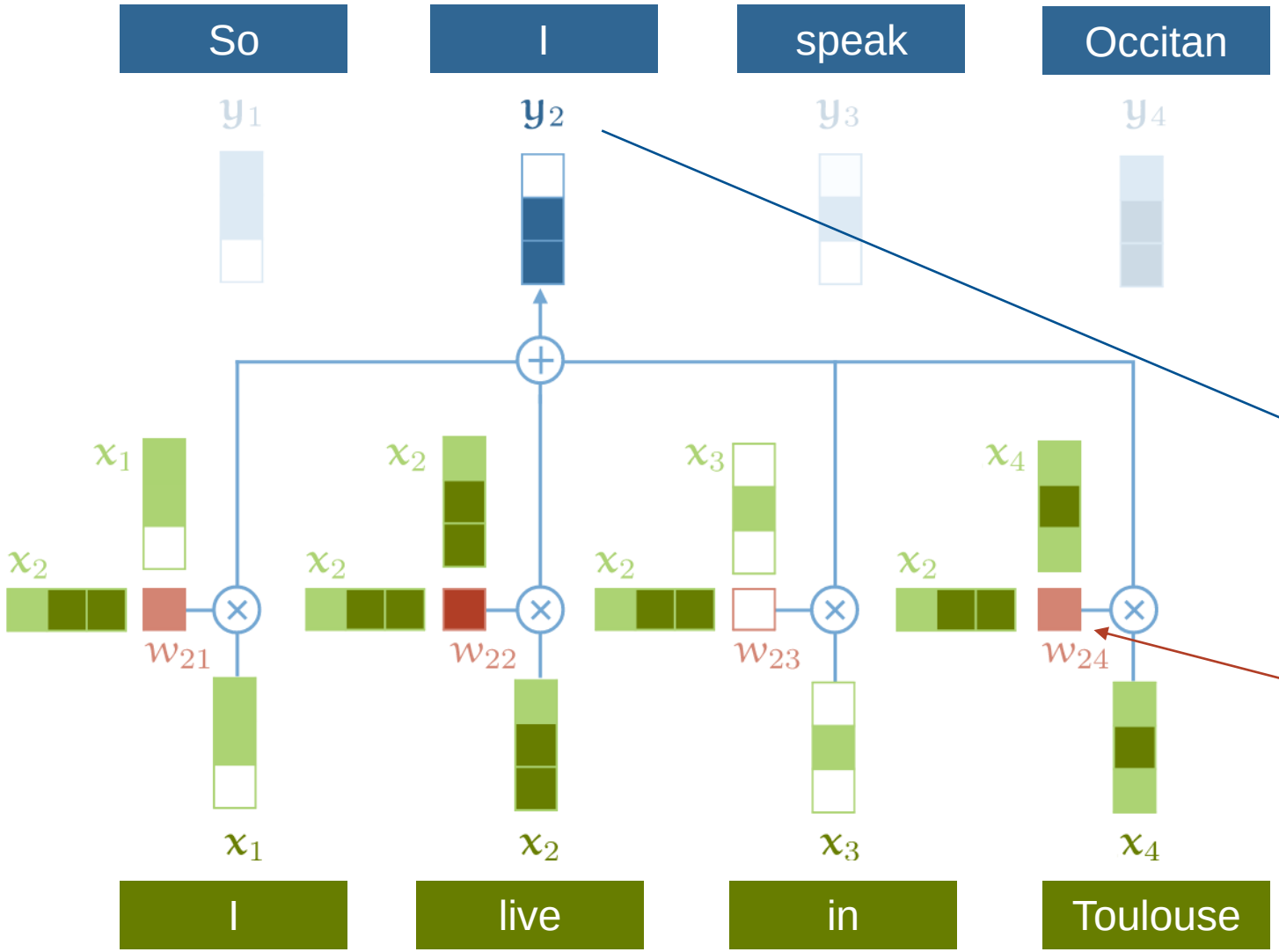
KyungHyun Cho Yoshua Bengio*
Université de Montréal

Cité 27180 fois

- Each of these recurrent outputs is kept in memory to form the final output.
- The Attention Encoder transmits much more information to the Decoder than in the classical approach



Deep Learning – Mécanisme d'attention et Transformers



Attention Is All You Need

2017 Cité 65053 fois

Ashish Vaswani* Google Brain avaswani@google.com
Noam Shazeer* Google Brain noam@google.com
Niki Parmar* Google Research nikip@google.com
Jakob Uszkoreit* Google Research usz@google.com
Llion Jones* Google Research llion@google.com
Aidan N. Gomez* † University of Toronto aidan@cs.toronto.edu
Łukasz Kaiser* Google Brain lukaszkaiser@google.com
Illia Polosukhin* ‡ illia.polosukhin@gmail.com

Google

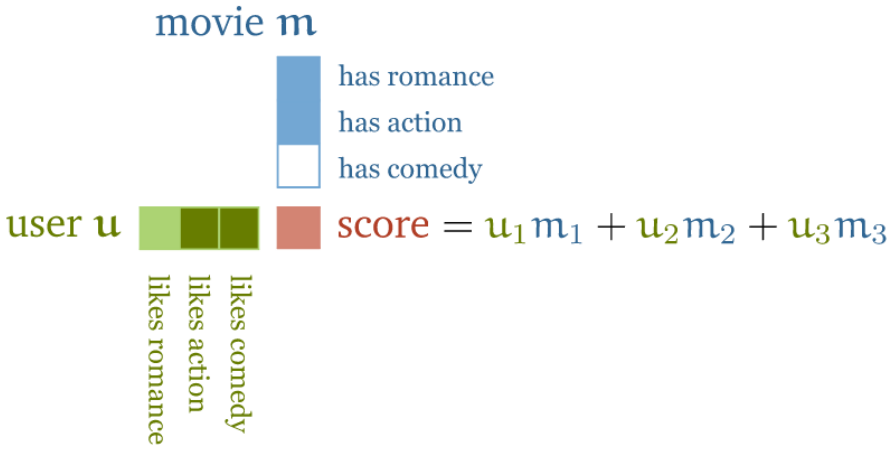
$$y_i = \sum_j \omega_{ij} x_j$$

$$\omega_{ij} = \frac{\exp \omega'_{ij}}{\sum_j \omega'_{ij}}$$

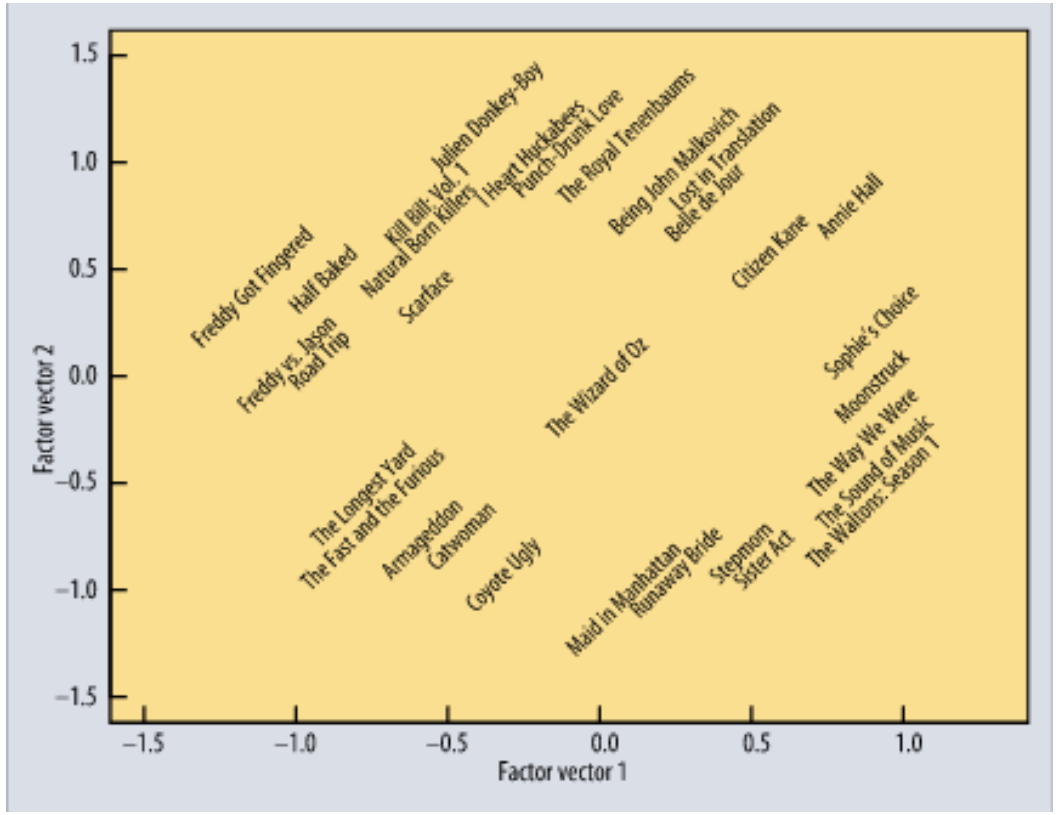
$$\omega'_{ij} = x_i^T x_j$$

Deep Learning – Attention mechanism and Transformers

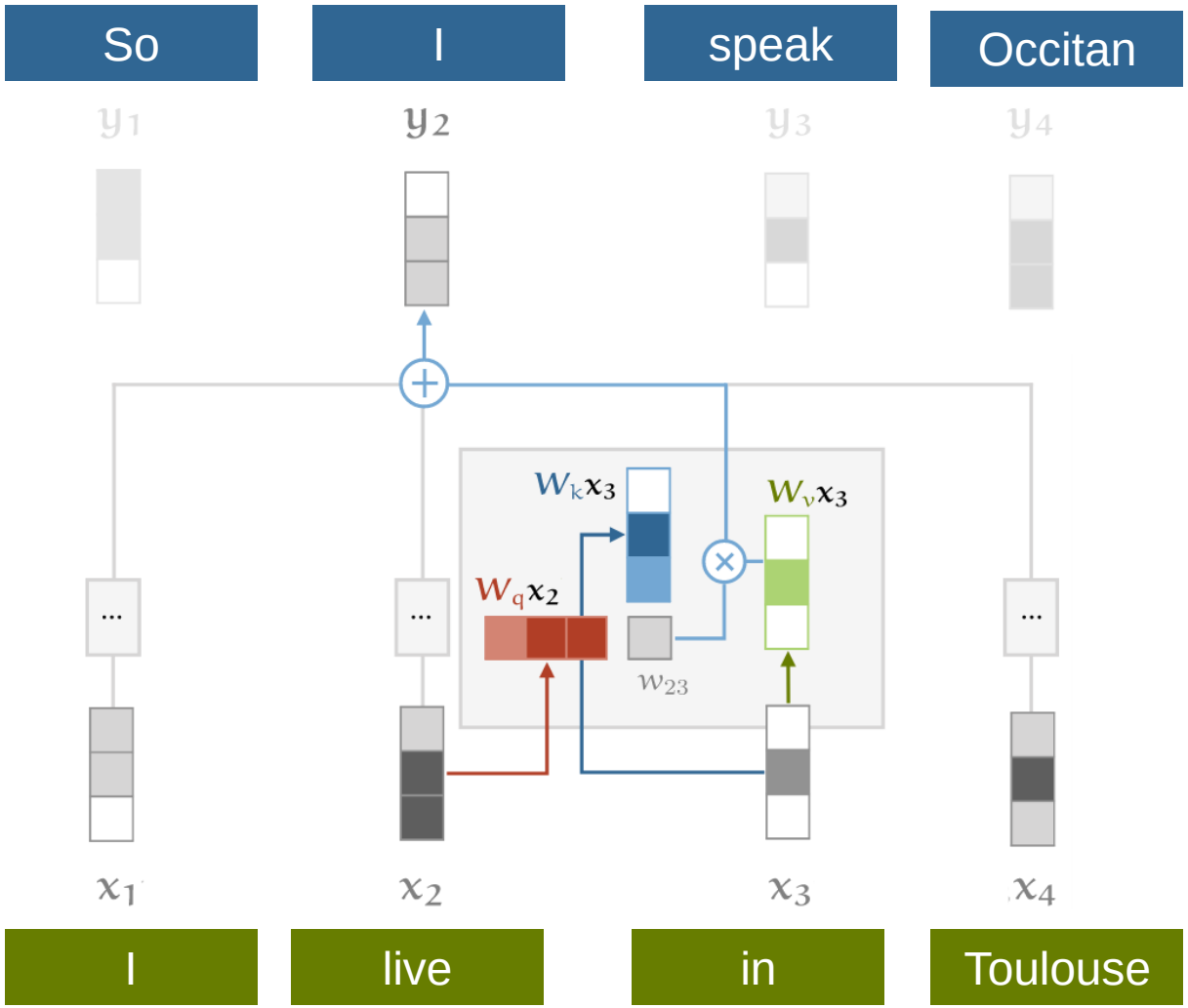
❖ **Example: Film recommandation**



Classical approach



Deep Learning – Attention mechanism and Transformers



- **Until now, no parameters to learn!**
- **Each input vector xi is used in three different ways in the auto-attention operation:**
  - It is compared to all the other vectors to establish the weights of its own output y_i → « request », matrix W_q
  - It is compared to all the other vectors to establish the weights of the output of the j-th vector y_i→ « key », matrix W_k
  - It is used as part of the weighted sum to calculate each output vector once the weights have been established. y_i → « value », matrix W_v

$$\mathbf{q}_i = \mathbf{W}_q \mathbf{x}_i \qquad \mathbf{k}_i = \mathbf{W}_k \mathbf{x}_i \qquad \mathbf{v}_i = \mathbf{W}_v \mathbf{x}_i$$

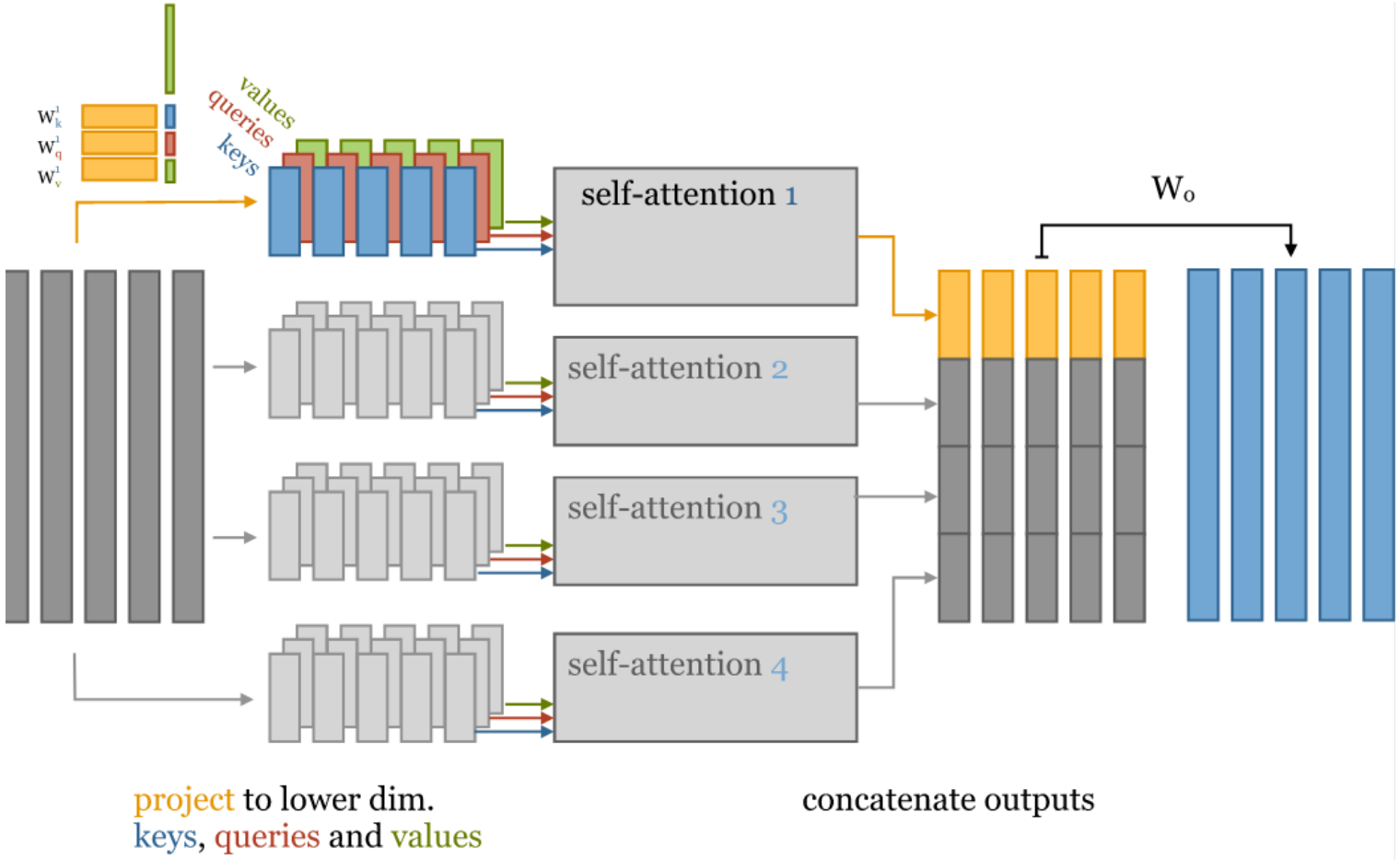
$$w'_{ij} = \mathbf{q}_i{}^T \mathbf{k}_j$$

$$w_{ij} = \text{softmax}(w'_{ij})$$

$$\mathbf{y}_i = \sum_j w_{ij} \mathbf{v}_j \,.$$

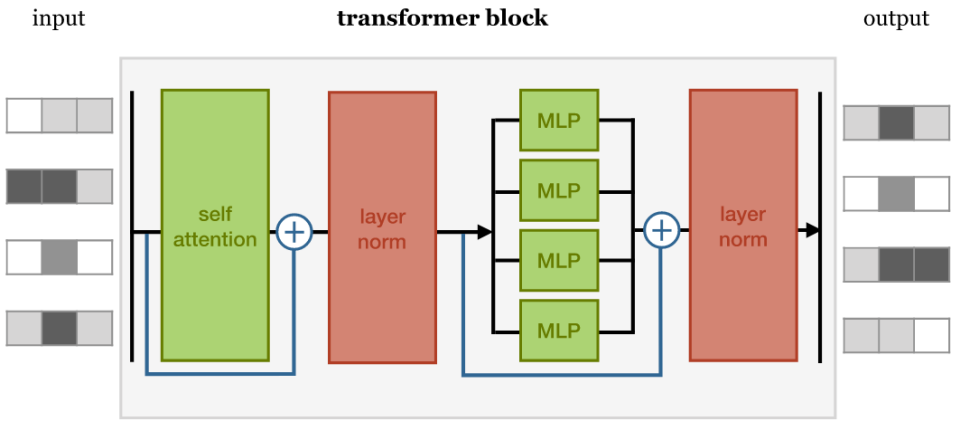
Deep Learning – Attention mechanism and Transformers

- **Problem : The same word appears in several context => Multi-head attention**

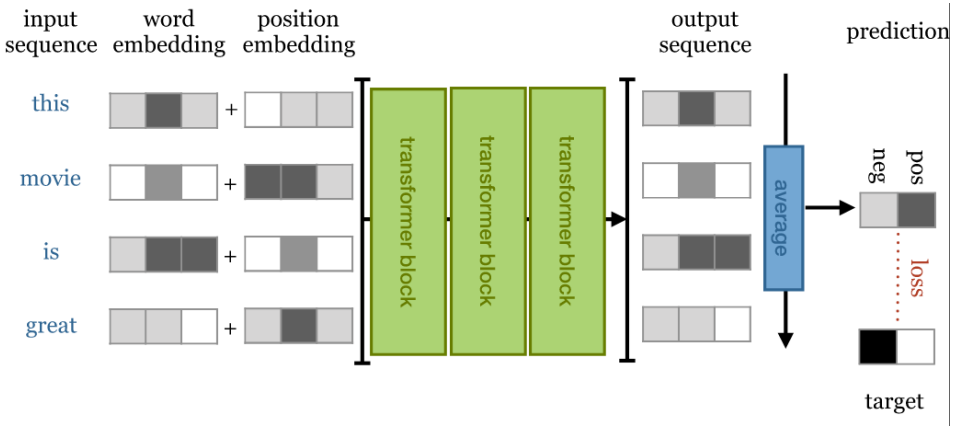


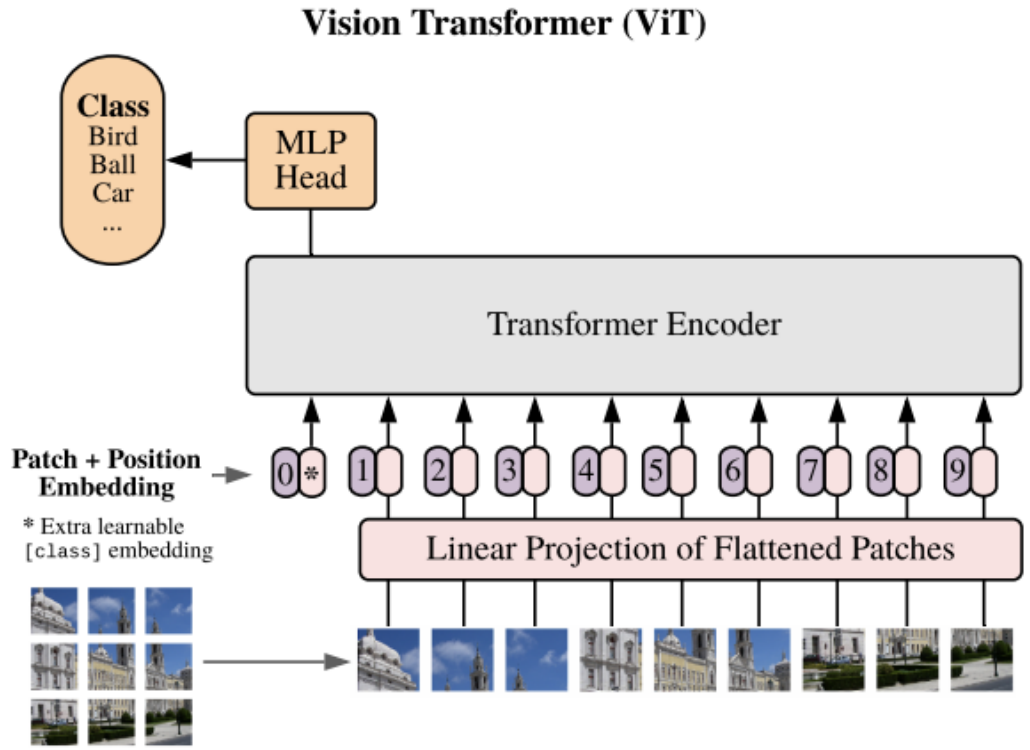
Deep Learning – Attention mechanism and Transformers

❖ Transformer block

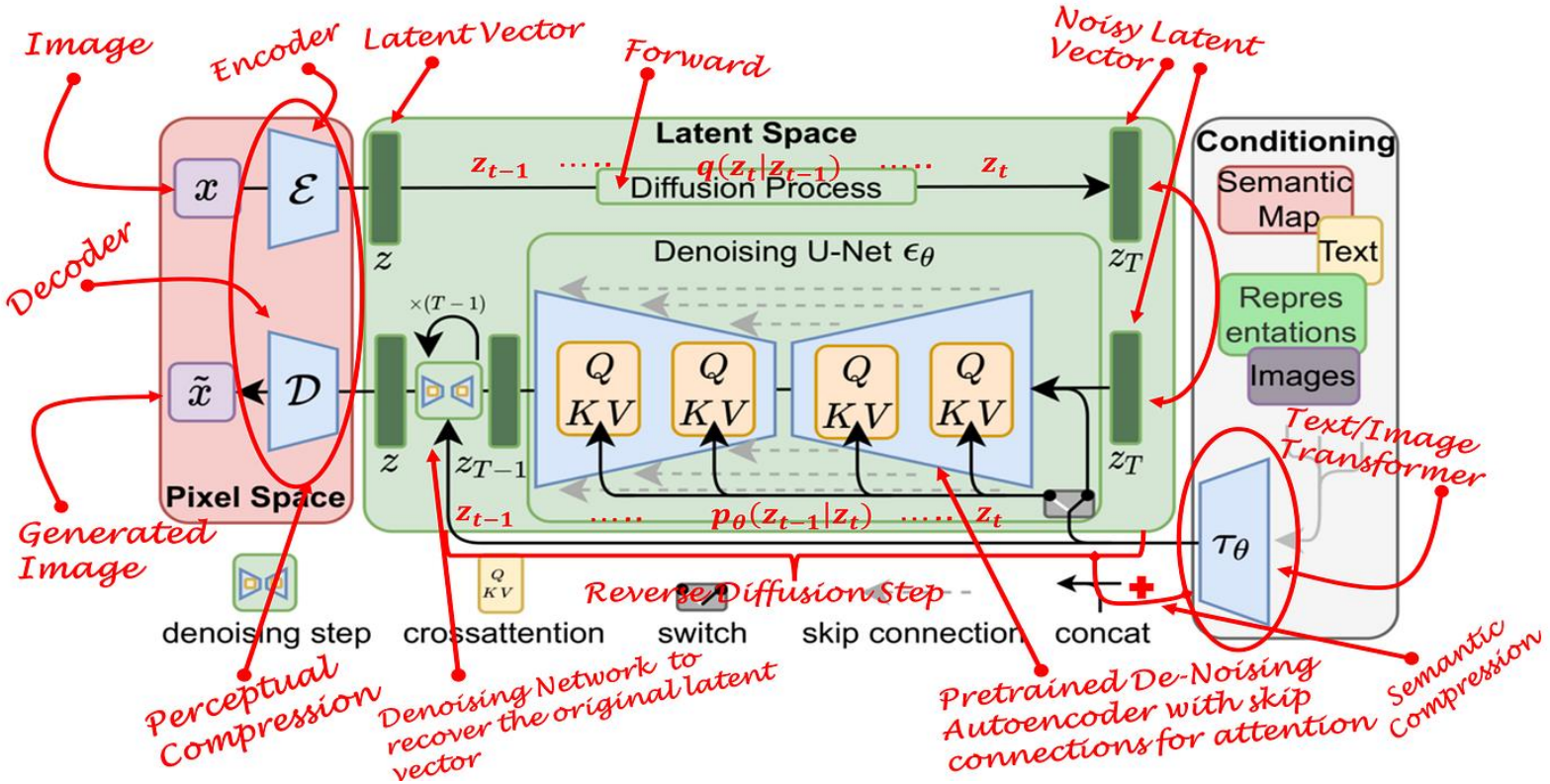


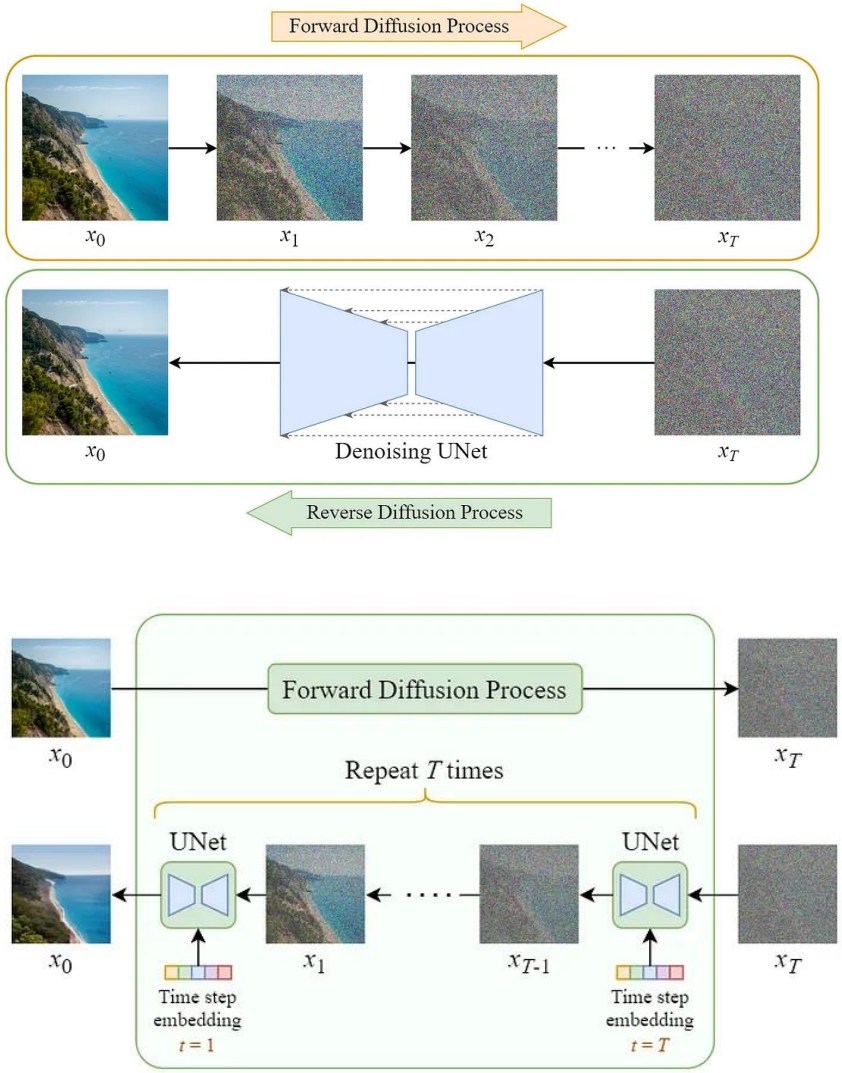
BERT, GPT-(2,3,4)…





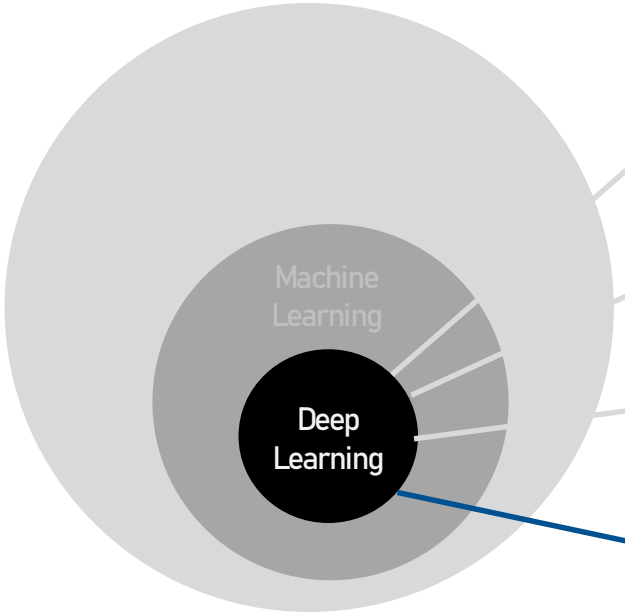
Deep Learning – Diffusion model





Stable Diffusion, MidJourney,…

Different DL models



Supervised

- Deep Neural Network (DNN)
- Convolutional Neural Network (CNN)
- Reccurent Neural Network (RNN)
- Long-Short Term Memory (LSTM)

Unsupervised

- Generative Adversarial Network (GAN)
- Recurrent Neural Network (RNN)
- Long-Short Term Memory (LSTM)
- Gated Recurrent Units (GRU)

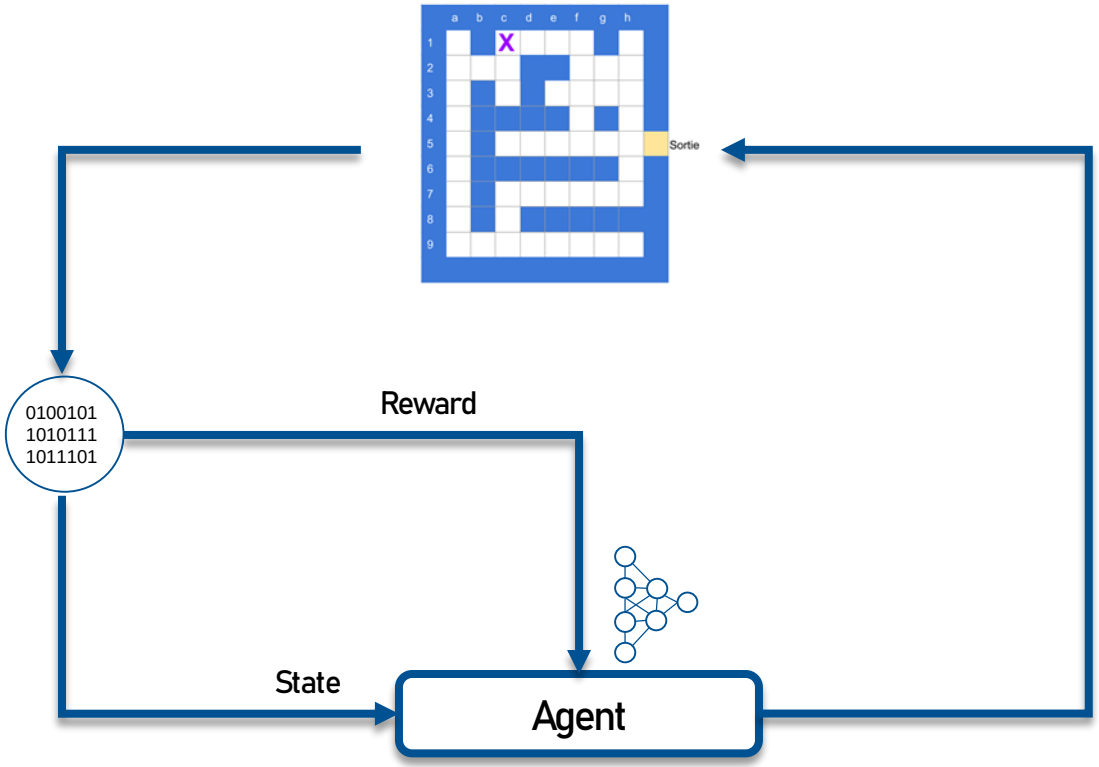
Semi-Supervised

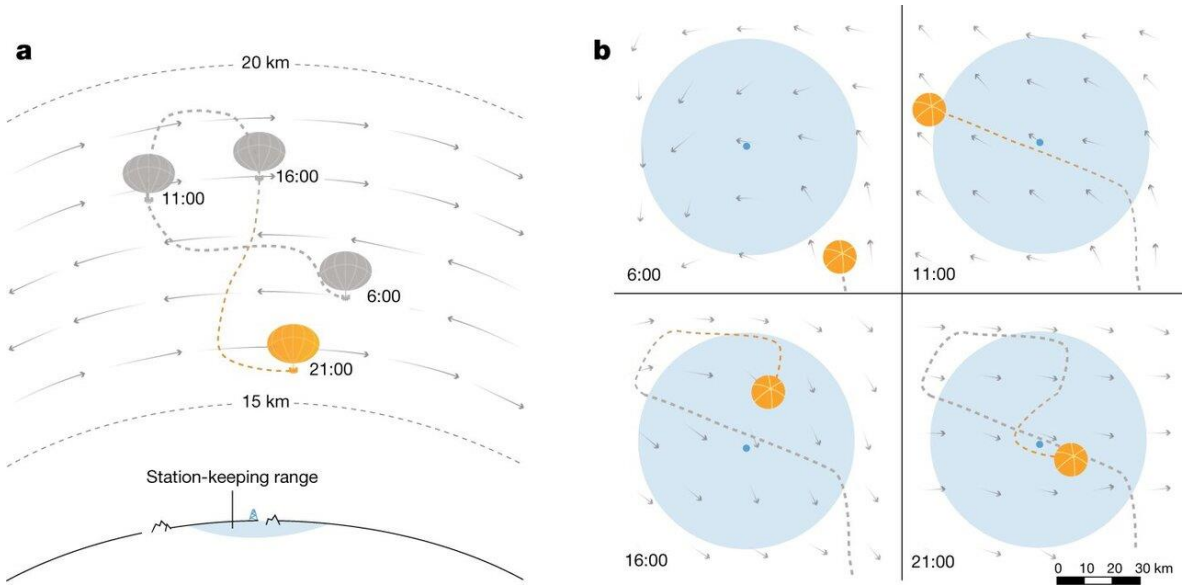
- Auto Encodeurs (AE)
- Restricted Boltzmann Machines (RBM)
- Deep Boltzmann Machines (RBM)

Reinforcement

- Deep Q Net (DQN)
- Deep Deterministic Policy Gradients (DDPG)
- Normalized Advantage Functions (NAF)

Reinforcment learning



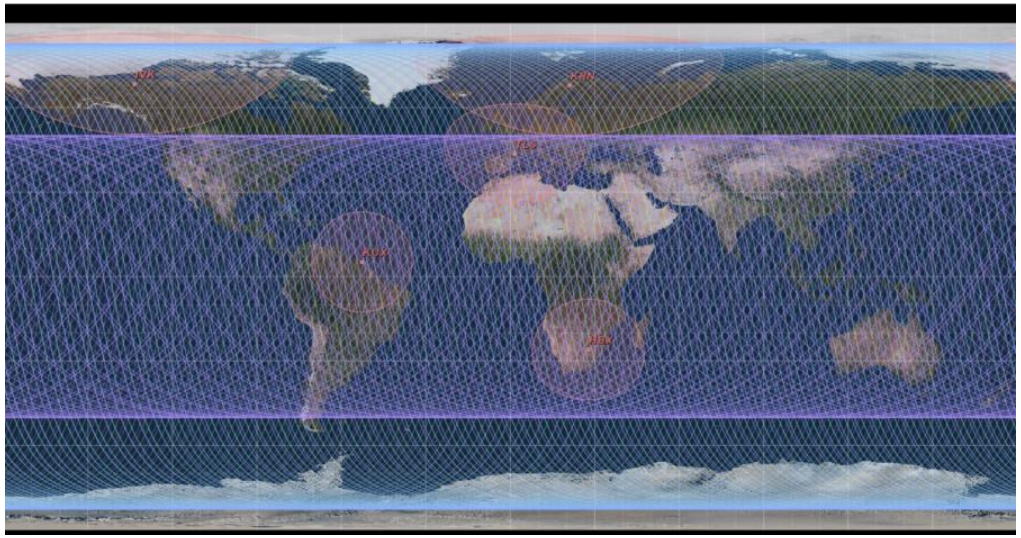


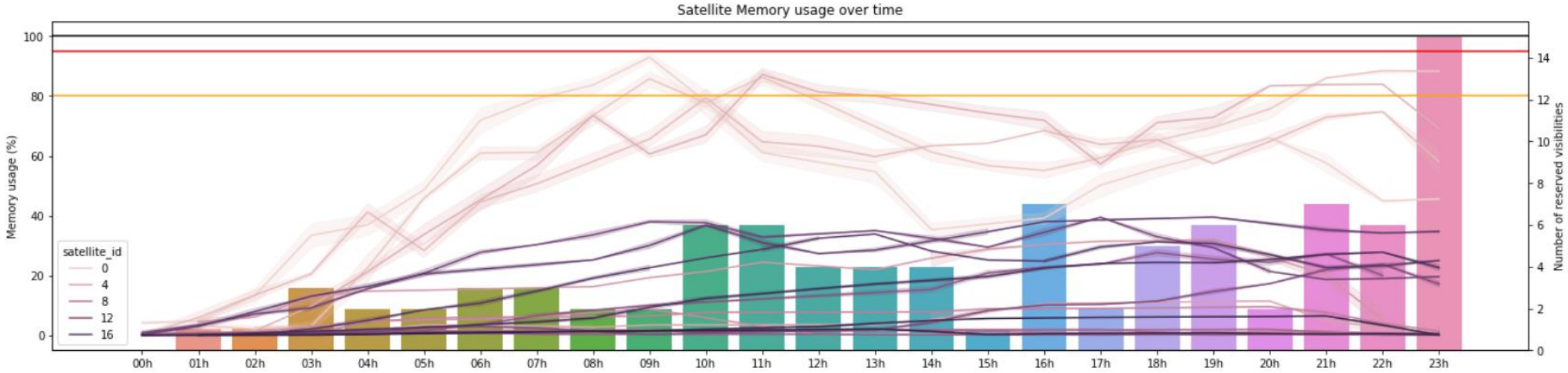
Example : Station Keeping

Utilisation : Game, Guided navigation, Recommandation system, ChatBot ...

Reinforcment – CNES example

- Name: PIVOS
- Objective : Dumping plan of a constellation's TM using Machine Learning
- Methods: Renforcement: DQN, PPO, MCTS
- Data: Simulation up to 3 constellations of 20 satellites + 5 stations





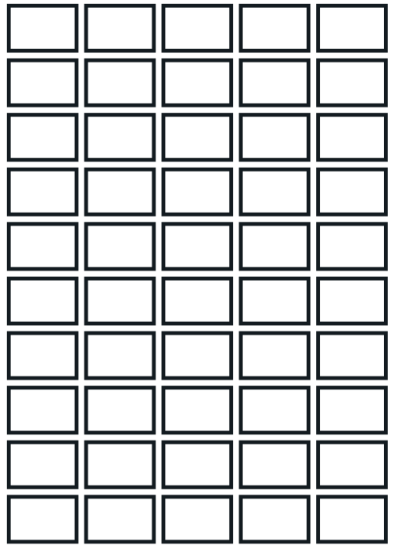
How to do ML/DL ?

Nota Bene on Machine Learning / Deep Learning ?

1. Machine learning requires a massive amount of data to train neural networks, which is not easy to obtain every time.
2. Selecting the right algorithm is crucial because the results can be biased and lead to inaccurate predictions.
3. It is difficult to converge algorithms (the hypothesis space is enormous).
4. Neural networks do not have the ability to generalize and are bound by their training data, meaning there is a lack of creativity and they are only effective in what they already know.
5. Deep learning is always a "black box" algorithm that is not explained.
6. How can we certify an AI model developed from a limited number of assumptions (inputs)?
7. There can be a deep gap between the resources available to develop a model and those used for inference.
8. Deep learning is an energy-intensive process, and its environmental footprint is not negligible

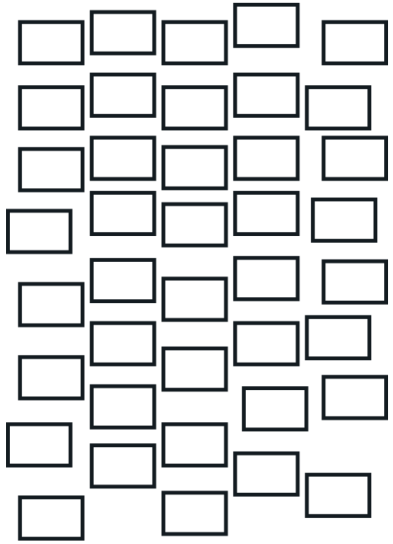
WORKFLOW IA – Data types

Structured



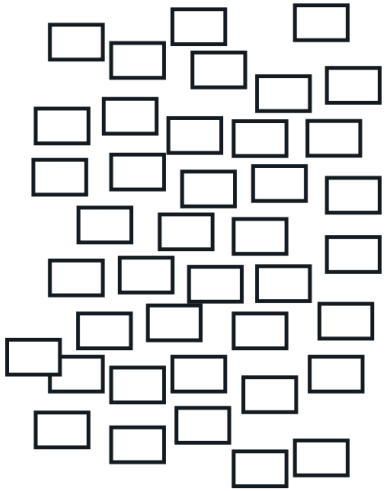
- ✓ Structure bien définie
- ✓ Conforme à un modèle de données
- ✓ Facilement accessible
- ✓ Généralement stockées sous forme de tableau
- ✓ Exemples:
  - Données classiques dans les bases traditionnelles: nombres, catégories, booléens...
  - Données géolocalisées avec au minimum des coordonnées spatiales
  - Données provenant de capteurs variés: signaux, télémétries...

Unstructured



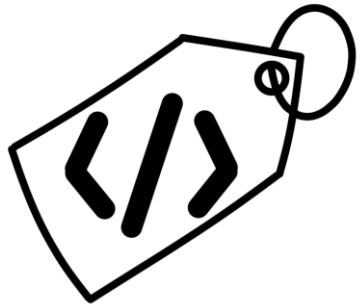
- ✓ Pas de format pré-défini ou organisation des données non structurées
- ✓ Beaucoup plus difficile à capturer, traiter et analyser
- ✓ Exemples:
  - Données textuelles provenant d'articles, de documents, de commentaires, d'e-mails...
  - Des images provenant d'instruments d'observation: visible, infrarouge, radar...

Not structured



- ✓ Ne sont pas stockées dans une base de données relationnelle comme les données structurées
- ✓ Certaines propriétés organisationnelles qui facilitent l'analyse.
- ✓ Exemples:
  - HTML, XML, JSON
  - Bases de données NoSQL, ...

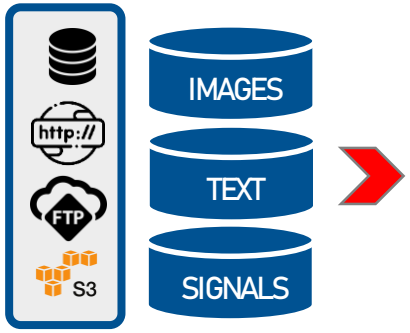
Metadata



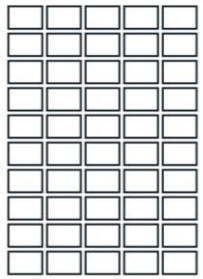
- ✓ « Les données sur les données »
- ✓ Les « métadonnées » permettent de classer, mesurer ou même documenter quelque chose relatif aux propriétés des données.
- ✓ Elles décrivent les informations pertinentes des informations.
- ✓ Exemples:
  - Métadonnées d'un document: auteur, taille du fichier, la date générés par le document, des mots clés pour définir le document, etc.
  - Informations de prise de vue, type de capteurs...

WORKFLOW IA – Data Selection

Data Selection

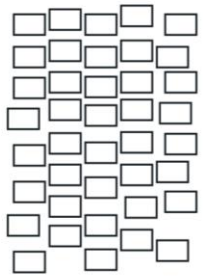


Données structurées



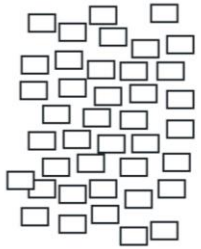
- ✓ Structure bien définie
- ✓ Conforme à un modèle de données
- ✓ Facilement accessible
- ✓ Généralement stockées sous forme de tableau.
- ✓ Exemples:
  - Données classiques dans les bases traditionnelles: nombres, catégories, booléens...
  - Données géolocalisées avec au minimum des coordonnées spatiales
  - Données provenant de capteurs variés: signaux, télémétries...

Données semi structurées



- ✓ Pas de format pré-défini ou organisation des données non structurées
- ✓ Beaucoup plus difficile à capturer, traiter et analyser
- ✓ Exemples:
  - Données textuelles provenant d'articles, de documents, de commentaires, d'e-mails...
  - Des images provenant d'instruments d'observation: visible, infrarouge, radar...

Données non structurées



- ✓ Ne sont pas stockées dans une base de données relationnelle comme les données structurées
- ✓ Certaines propriétés organisationnelles qui facilitent l'analyse.
- ✓ Exemples:
  - HTML, XML, JSON
  - Bases de données NoSQL, ...

Métadonnées

- ✓ « Les données sur les données »
- ✓ Les « métadonnées » permettent de classer, mesurer ou même documenter quelque chose relatif aux propriétés des données.
- ✓ Elles décrivent les informations pertinentes des informations.
- ✓ Exemples:
  - Métadonnées d'un document: auteur, taille du fichier, la date générés par le document, des mots clés pour définir le document, etc.
  - Informations de prise de vue, type de capteurs...

1

The data can be:
- on digital platforms:
- on a local file system,
- on an internal server,
- on a remote server,
- on the cloud.
- on sensor infrastructures.

Need for the field expert:
- What data is available?
- Are the data relevant?
- Are the data reliable?

Need for the data scientist:
- What is the volume of data?
- How to avoid biases?

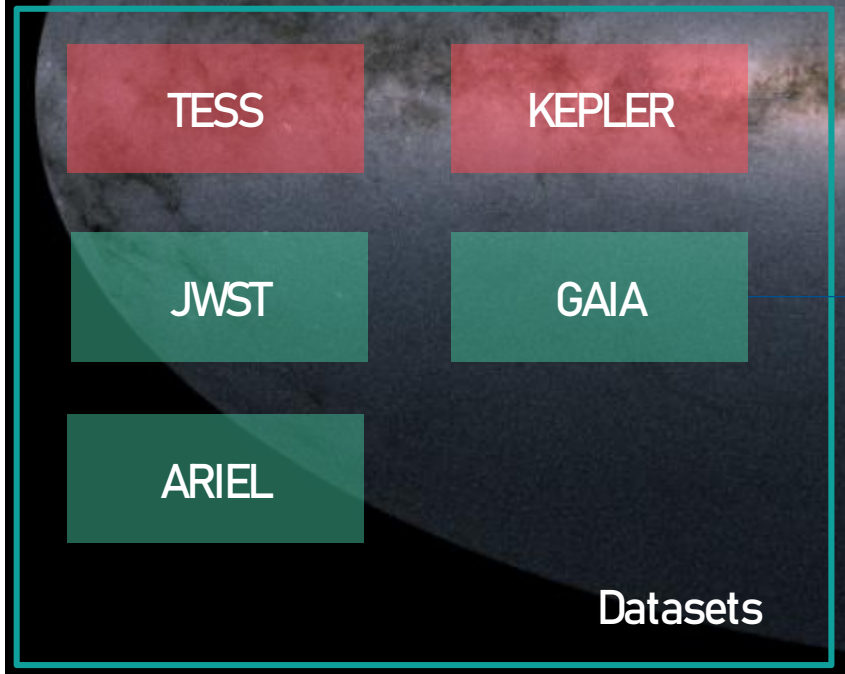
Working on a duo

WORKFLOW IA – Data Selection

Questions about data arise from the beginning of a project:

- Which dataset to use ?
- Is there enough instances ?
- Which labelisation to use with those data ?

For example, classification of spatial objects :



Using astrometric, photometric, and spectrometric data?

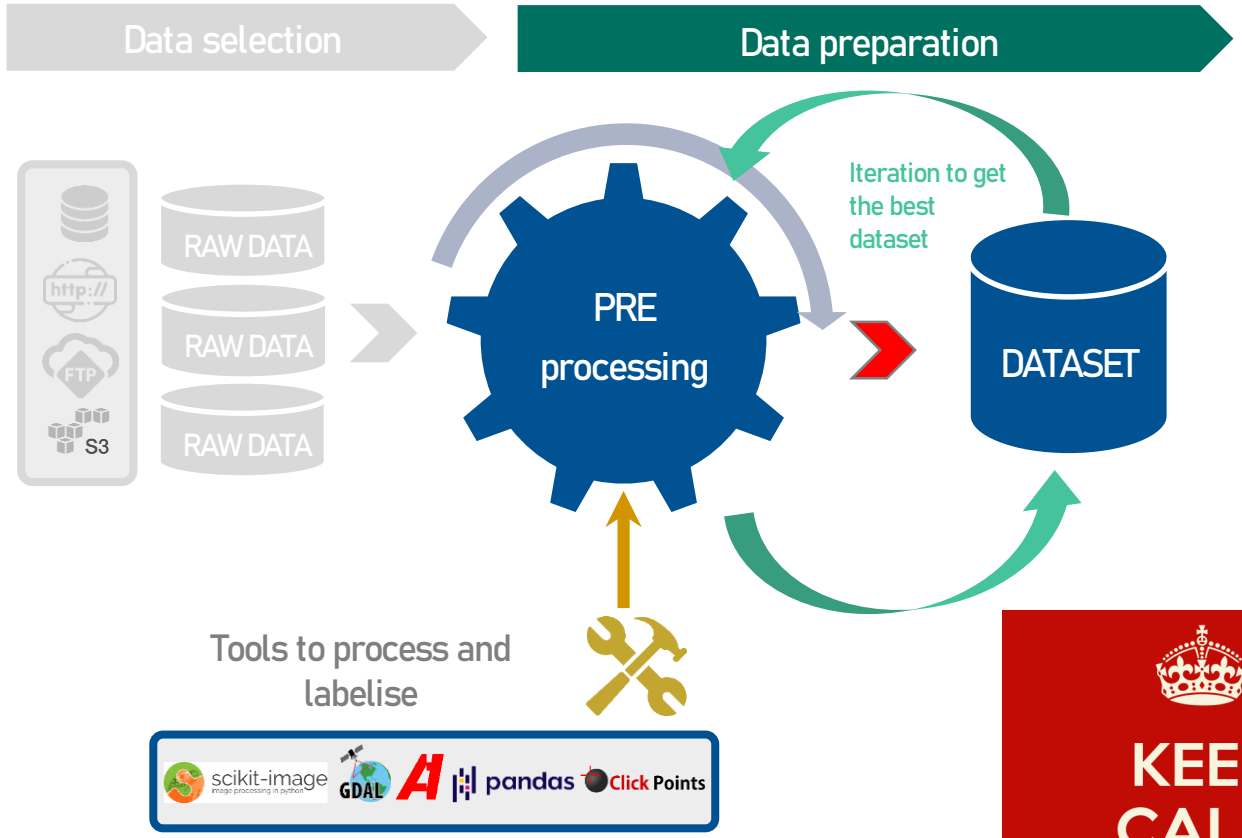
Way more stars than galaxies...

Which information to use ?

To cross-reference information with other catalogs to retrieve additional information?

Transforming the information, up to which point ?

WORKFLOW IA – Data preparation



2

Data preparation:

- Cleaning the data: duplicates, absurd or exceptional data (outliers), etc.
- Handling missing data
- Eliminating redundant attributes
- Possible normalization
- Discretizing attributes or conversely making discrete values continuous
- Data augmentation
- Generating synthetic data
- Varying data sources
- Labeling!

- ✓ Are the data free of rights?
- ✓ Are the data confidential, limited to restricted distribution?
- ✓ Are the data of a personal nature?

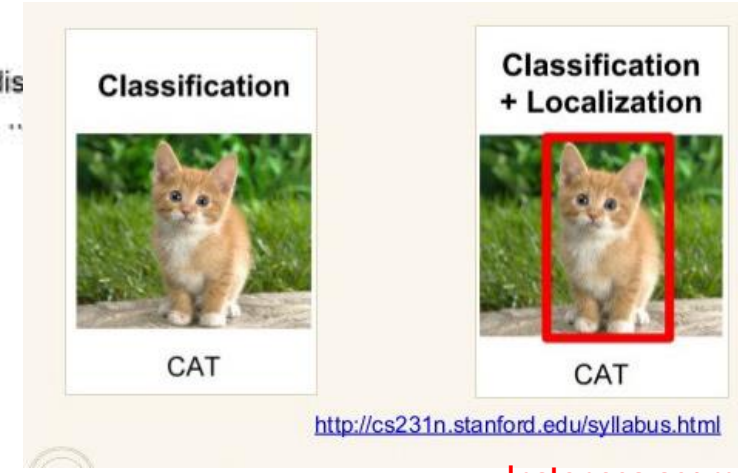
WORKFLOW IA – Preparing the data – Labelisation

Classification

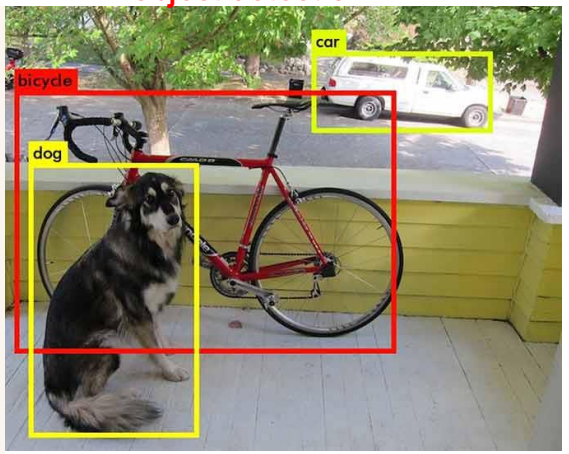
(assume given set of dis
{dog, cat, truck, plane, ..

cat

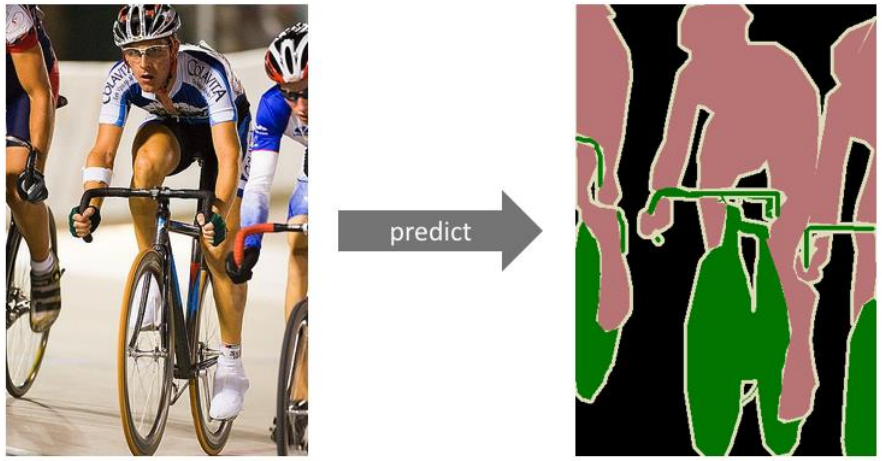
Classification + localization



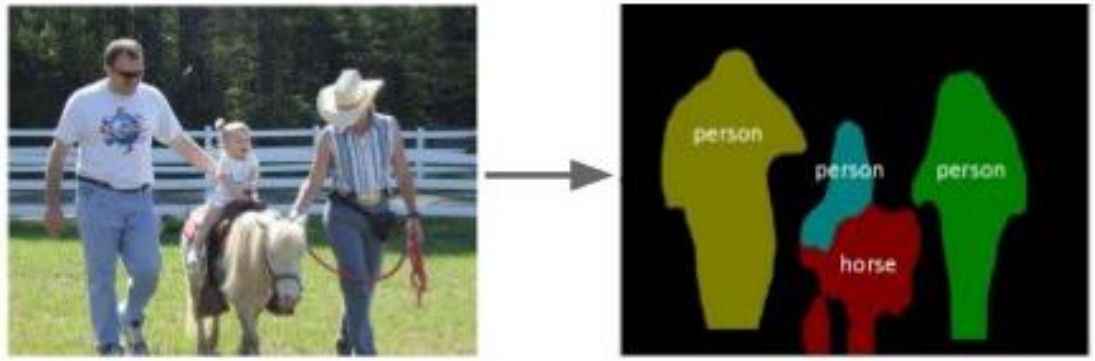
Object detection



Semantic Segmentation



Instances segmentation



Very (very) time consuming + need expert knowledge !

WORKFLOW IA – Preparing Data – Data Volumetry

The necessary quantity of data is relative to the complexity of the problem.

- The more data possible
- Never less than one hundred instances
- More generally, thousands, ten of thousands to use every AI models

More features ?
More parameters ?
More complex ?

→ Then more data requried

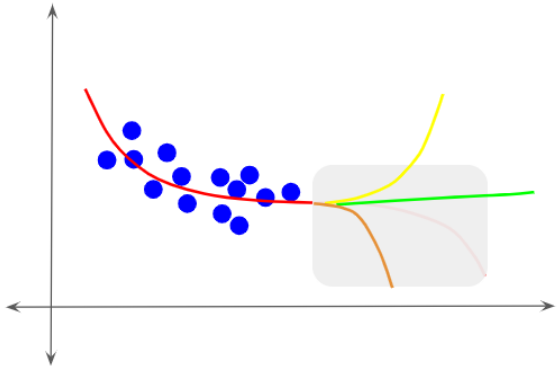
WORKFLOW IA – Data preparation – Quality or Quantity ?

Attention !

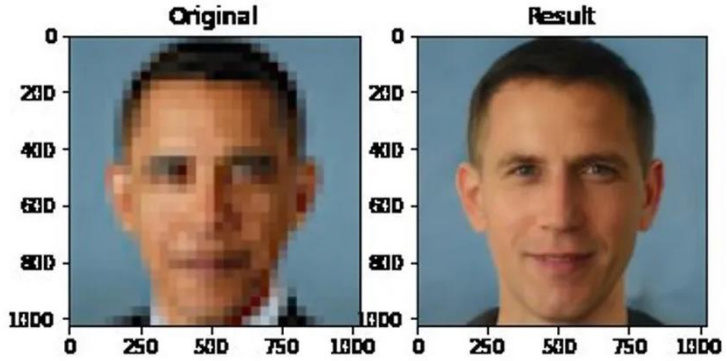
- **Inaccurate, incomplete, or mislabeled data:** The information is either bad or has not been properly cleaned. How to analyze terabytes or even petabytes of information?
- **Having too much data:** The mass of information does not indicate their quality and relevance to a specific use case. All this extra data can cause "noise" that can influence the results of a model. This could make it difficult for the model to "generalize" its learning.
- **Having too little data:** Training a model on a small set can produce acceptable results in a test environment. The model may be unable to handle information flows in production and may generate biased scores.
- **Biased data:** Biases are multiple. If they are not bad in themselves, some data sampled from a larger set may not properly represent it.
- **Unbalanced data:** Unbalanced sets can significantly hinder the performance of machine learning models. They cause over-representations of one community or group while diminishing the importance of another. Example: bank fraud.
- **Data silos:** Only certain groups have access to certain data. Not all "available" data is used.
- **Inconsistency of data:** duplication of the same data with different values (e.g., different treatment for a given group).

WORKFLOW IA – BIASES

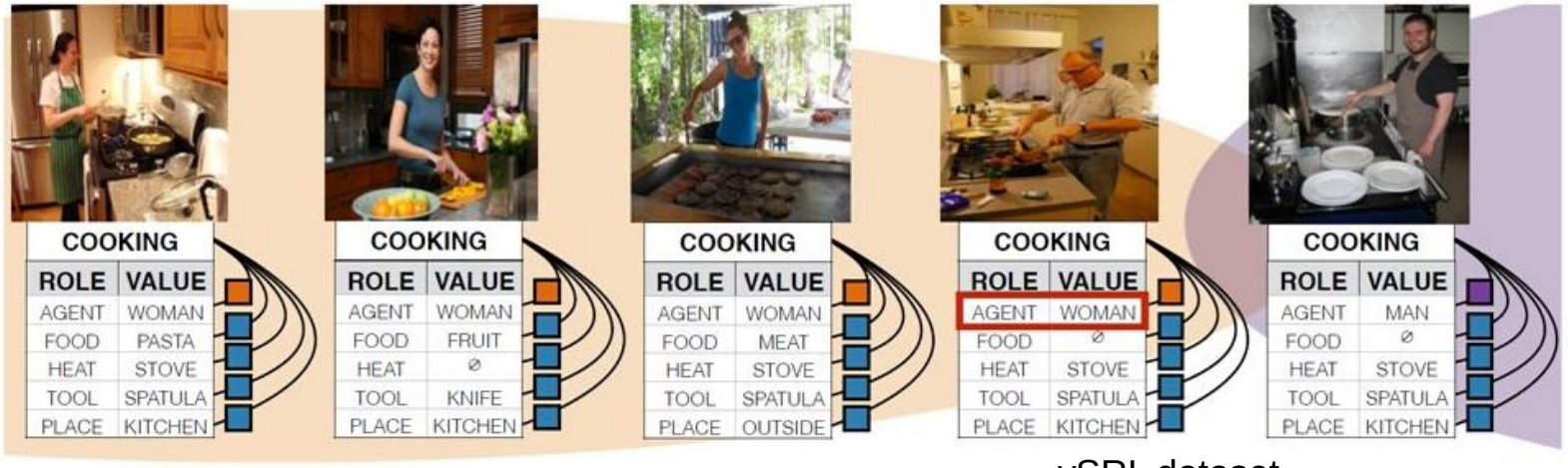
- **Bias on individuals**: data is poorly labeled by humans
- **Prejudice bias**: adding a bias to the data
- **Confirmation bias**: manipulating the data towards a hypothesis
- **Class balance bias**: When there is too much data from certain groups



Covariate shift



The PULSE algorithm takes pixelated faces and turns them into high-resolution images. Duke University. Biais discriminatoire et raciste

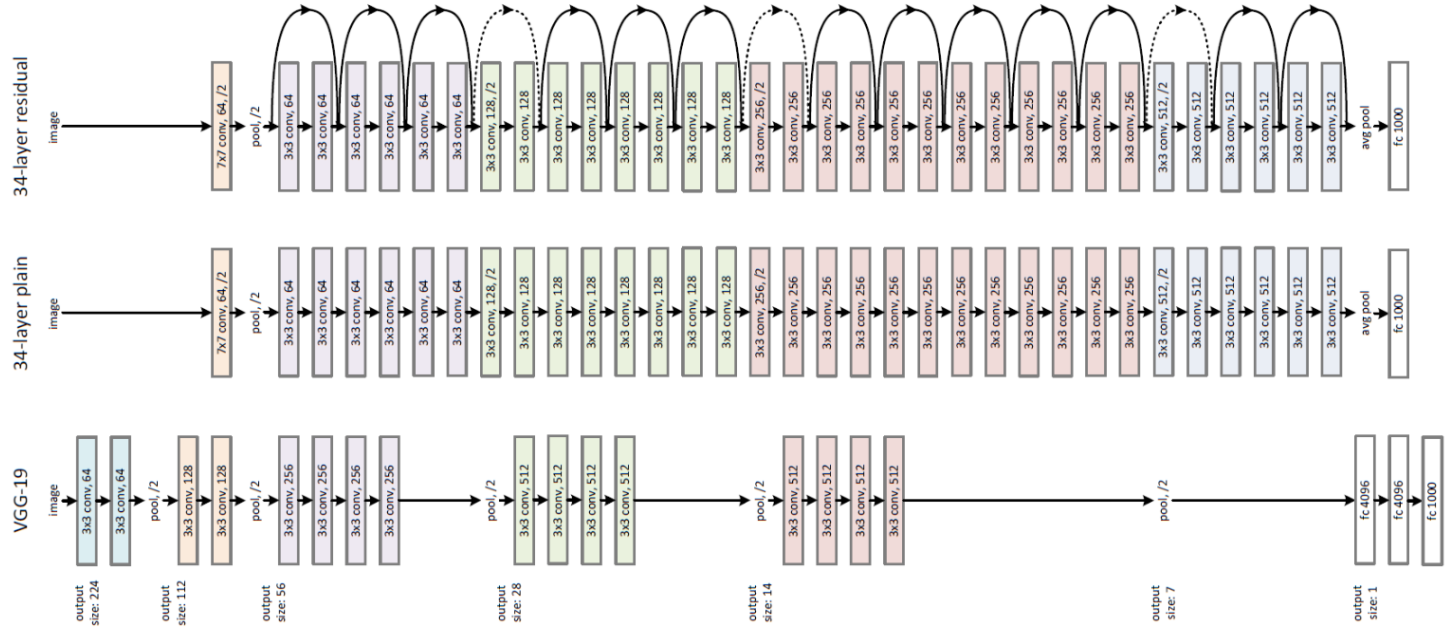


vSRL dataset

Solutions:

- Labeling control
- Completing information
- Multiple points of view
- Creation of artificial population / reduction

WORKFLOW IA – Deep Learning = Big Data and Big CNN



ImageNet:
~14 millions images
~21000 classes
https://devopedia.org/imagenet

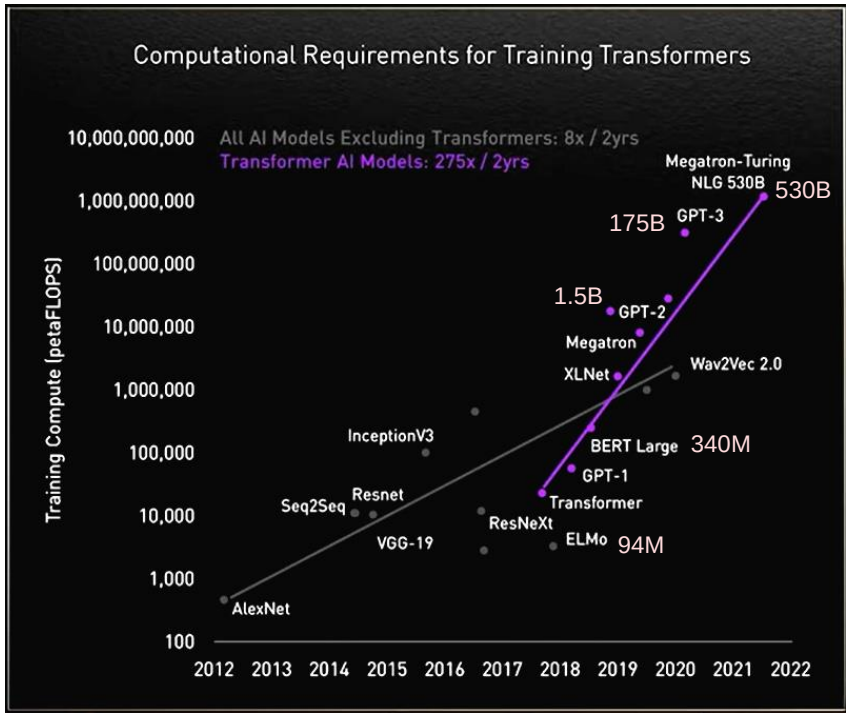
Resnet152:
Network with 152 layers
~60M parameters
~11 billions of FLOPS
https://towardsdatascience.com

In 2016

WORKFLOW IA – Deep Learning = Big Data and Big CNN

Learning diffusion models (ex DALL-E 2, MidJourney, Stable Diffusion…):

- 12 BILLIONS parameters
- 256 GPU A100
- 150 000 GPUh
- Cost ~600 000 $



Images générées par MidJourney v5

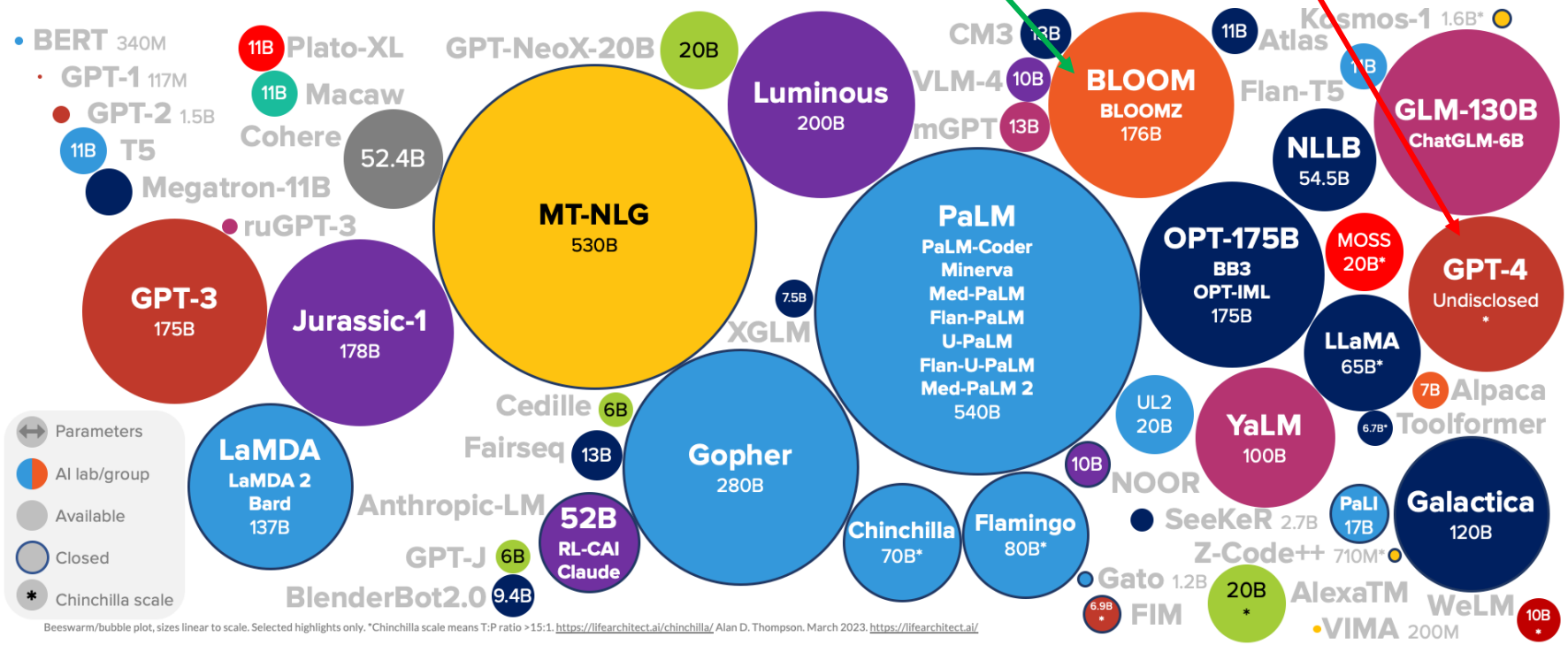
https://twitter.com/TechSpatiales/status/1640422959582986264

En 2023

2020 / 2023 - La course à l'échalotte

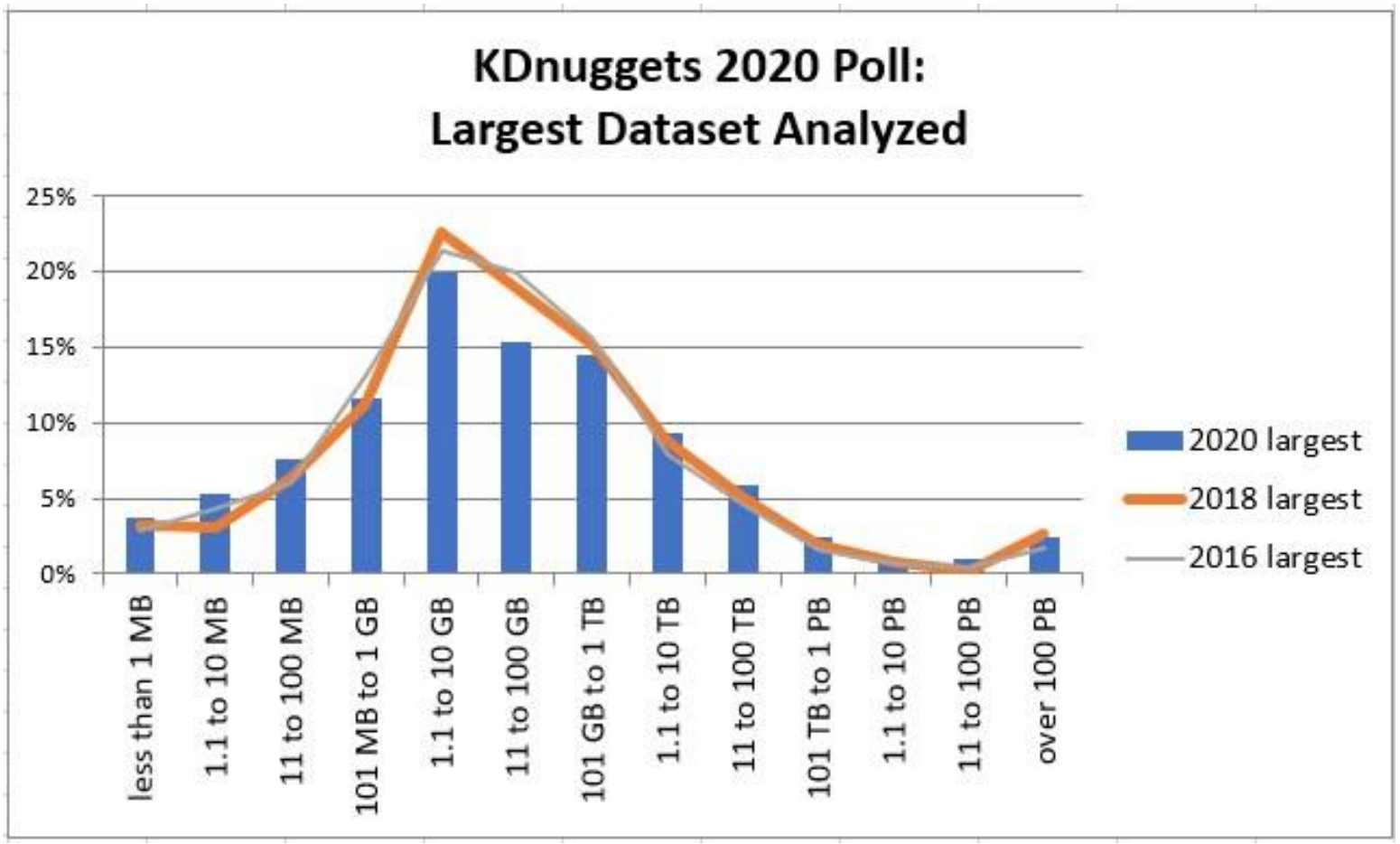
- Google: LaMDA (137B, May 2021), and PaLM (540B, Apr 2022)
- Meta: OPT (175B, May 2022), and BlenderBot 3 (175B, Aug 2022)
- DeepMind: Gopher (280B, Dec 2021), and Chinchilla (70B, Apr 2022)
- Microsoft-Nvidia: MT-NLG (530B, Oct 2021)
- BigScience: BLOOM (176B, June 2022)
- Baidu: PCL-BAIDU Wenxin (260B, Dec 2021)
- Yandex: YaLM (100B, June 2022)
- Tsinghua: GLM (130B, July 2022)
- AI21 labs: Jurassic-1 (178B, Aug 2021)
- Aleph Alpha: Luminous (200B, Nov 2021)

No official info on GPT-4, some are saying 1.76 TRILLION !



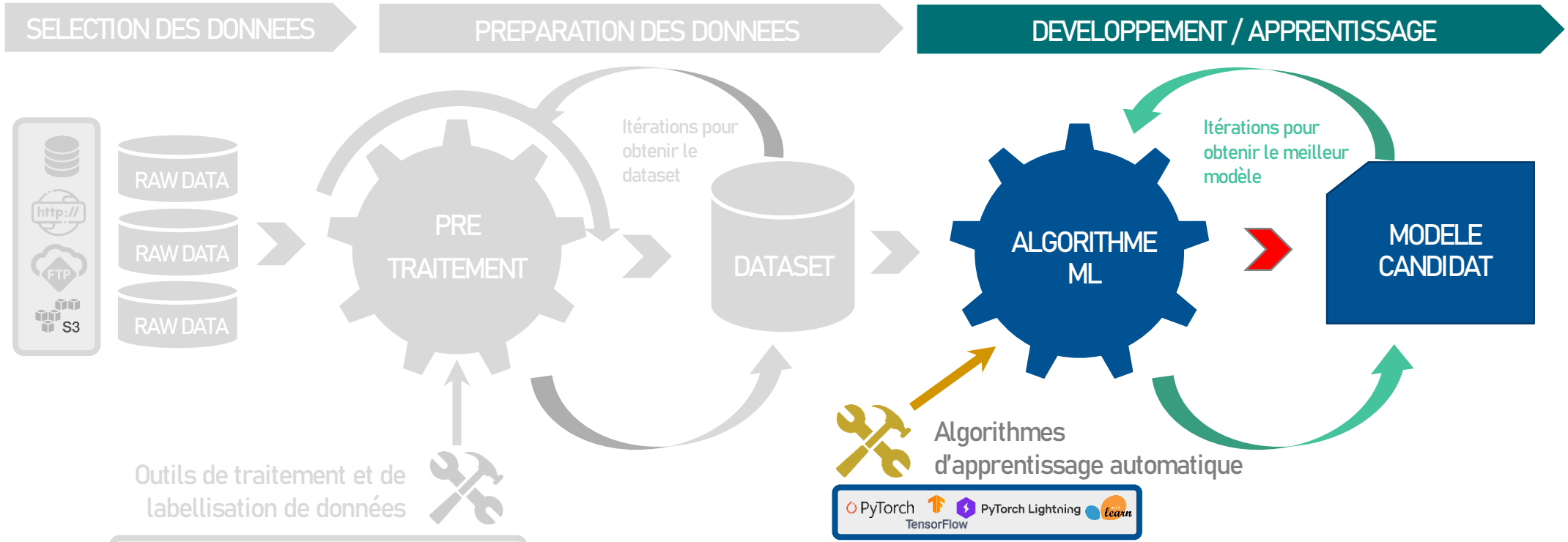
https://lifearchitect.ai/models/

WORKFLOW IA – Deep Learning = Big Data (dans la vraie vie)



KDnuggets Poll: Largest Dataset Analyzed, 2020, 2018, 2016
https://www.kdnuggets.com/2020/07/poll-largest-dataset-analyzed-results.html

WORKFLOW IA – Développement / Apprentissage



3

Cœur de métier du data scientist:

- Choix des architectures
- Choix des initialisations
- Choix des mesures d'erreurs
- Choix des métriques de performance
- Retravailler les données

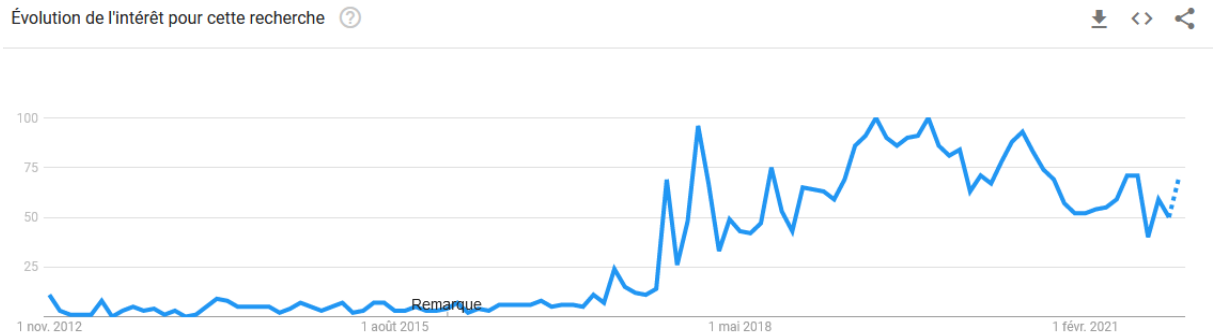
- Infrastructure de calcul
- Gestion des expériences
- Reproductibilité
- Cohérence avec les contraintes de déploiement ?

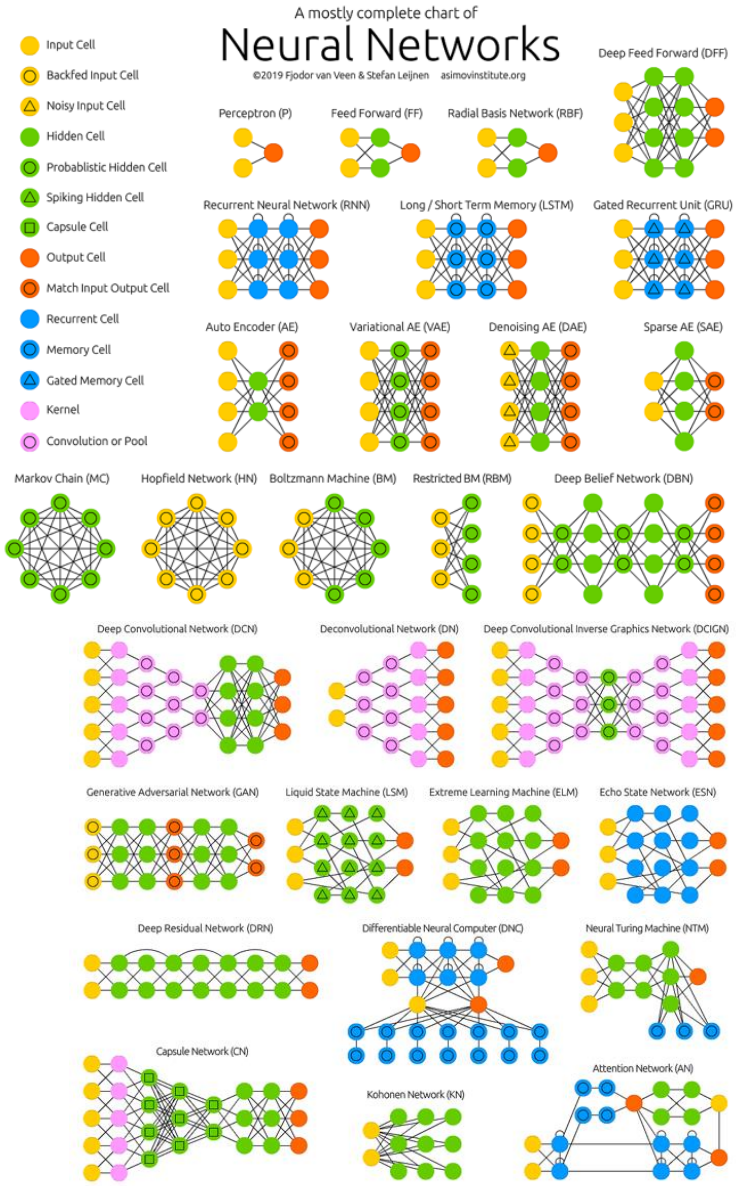
WORKFLOW IA – Développement – Le bon modèle

- **Théorême du « No Free Lunch »** (pas de déjeuner gratuit):

  En essence, ce théorème statue **qu'aucun modèle ou algorithme ne fonctionne bien pour *tous* les problèmes**. En d'autres termes, si un algorithme de machine learning fonctionne bien sur un type de problème particulier, ça veut dire qu'il le paiera ailleurs et sera donc moins performant en moyenne sur le reste des problèmes.

- Il faut trouver la bonne architecture pour le problème abordé
  - Définir le nombre de couches
  - Définir le type de couches
  - Définir le type d'activation en sortie des couches
  - Définir les relations entre les couches
- Tout un pan de la recherche en IA: **AutoML**
- Il s'agit d'automatiser la création de ces modèles de Machine Learning.





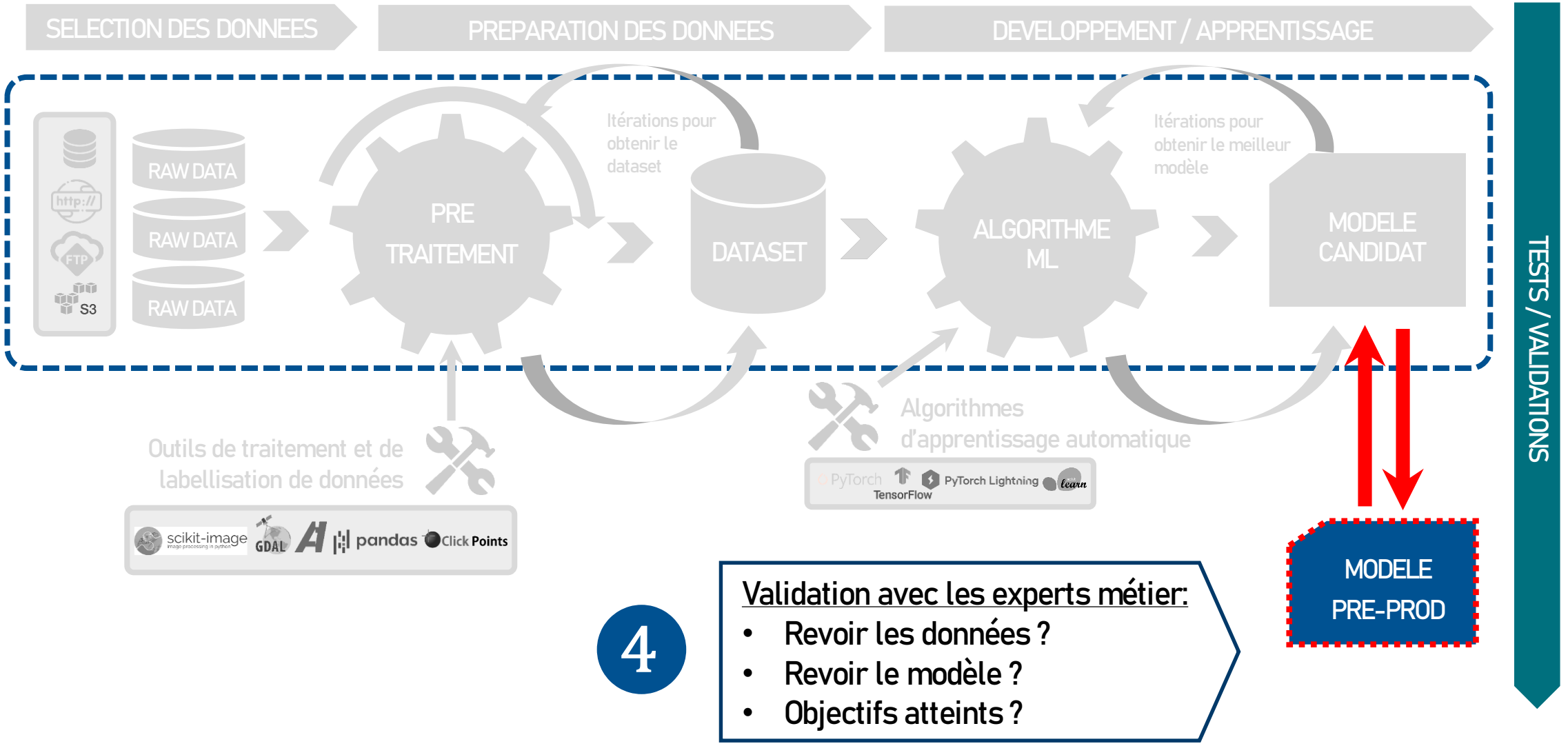
WORKFLOW IA – Développement – Attention à l'absence de sens commun / de généralisation

« Toute formule, si vaste soit-elle, impuissante à étreindre une Diversité qui n'a pas de limites, perd fatalement toute signification quand on s'écarte beaucoup des conditions où notre connaissance s'est formée », J.-H. Rosny

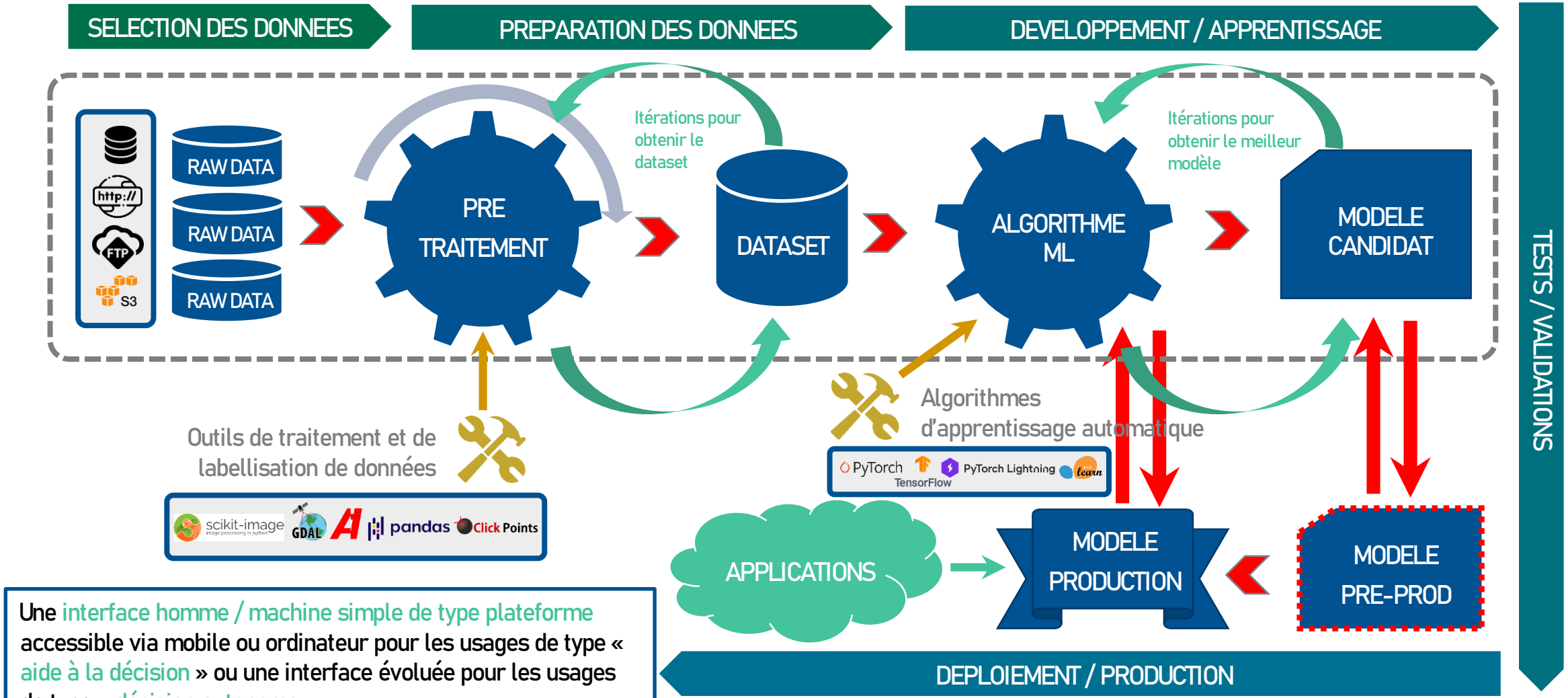
- Nos algorithmes se trompent parfois là où un humain ne commettrait pas d'erreur.
- Les réseaux de neurones reconnaissent des objets mais ne savent pas les identifier. Ils ont appris des listes de noms, savent coller des étiquettes sur les objets, sans les comprendre.
- A la différence, l'humain comprend ce qu'il voit, d'un point de vue sémantique.
- Exemple:
  - L'humain peut éventuellement confondre 2 animaux, mais pas un chat et une cafetière, car ce sont deux domaines totalement différents.
  - Pour un algorithme d'IA, en revanche, il n'y a pas plus de différence entre un chat et une cafetière qu'entre un chat et un chien.
- Performants pour des tâches simples et répétitives, les systèmes d'IA n'ont en revanche aucune compréhension du monde qui les entoure, ou du contexte dans lequel ils opèrent:
- Exemple:
  - Après les attentats de Londres en juin 2017, l'algorithme de tarification dynamique d'Uber avait commencé par doubler le tarif des courses dans la zone concernée pour répondre à l'afflux de demandes.

La capacité de généralisation est restreinte à son domaine d'apprentissage.

WORKFLOW IA – Tests & Validations



WORKFLOW IA – Déploiement & Production

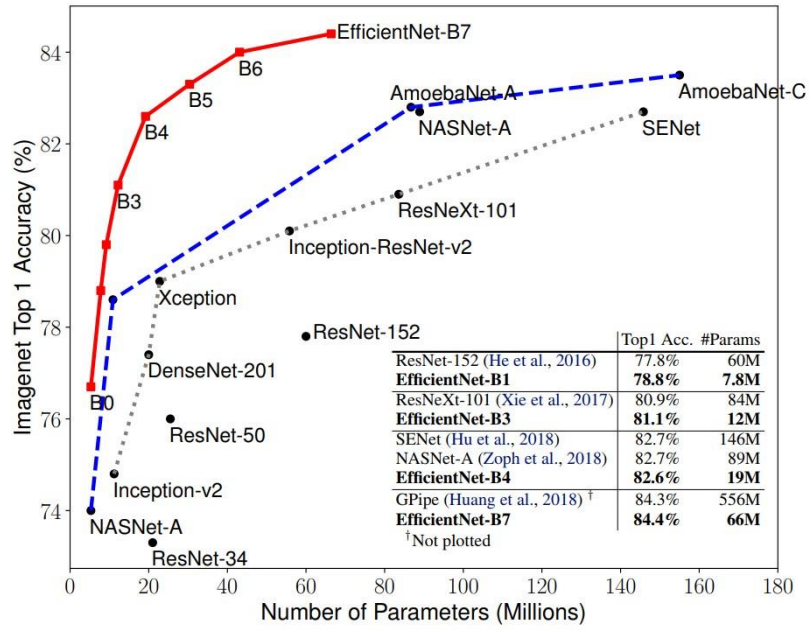


Une interface homme / machine simple de type plateforme accessible via mobile ou ordinateur pour les usages de type « aide à la décision » ou une interface évoluée pour les usages de type « décision autonome ».

WORKFLOW IA – Inference and boarding

- Although most developments will be made on HPC computing resources, their inference will have to run on low-energy, computational, and sometimes low-storage means.
- It is necessary to migrate the models to embedded means (mobile, FPGA, Raspberry PI, etc.), and ensure low-energy inference costs.

La recherche

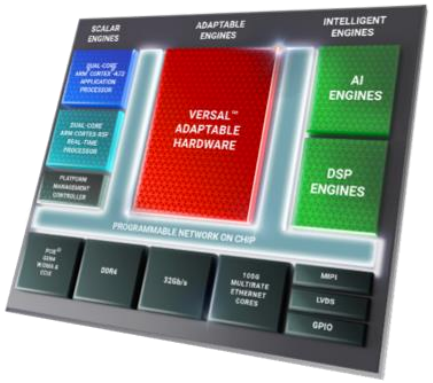


	Top1 Acc.	#Params
ResNet-152 (He et al., 2016)	77.8%	60M
EfficientNet-B1	**78.8%**	**7.8M**
ResNeXt-101 (Xie et al., 2017)	80.9%	84M
EfficientNet-B3	**81.1%**	**12M**
SENet (Hu et al., 2018)	82.7%	146M
NASNet-A (Zoph et al., 2018)	82.7%	89M
EfficientNet-B4	**82.6%**	**19M**
GPipe (Huang et al., 2018) †	84.3%	556M
EfficientNet-B7	**84.4%**	**66M**

†Not plotted

Surface Pro 9 avec Neural Processor Unit (NPU)

Qualcomm AI Engine

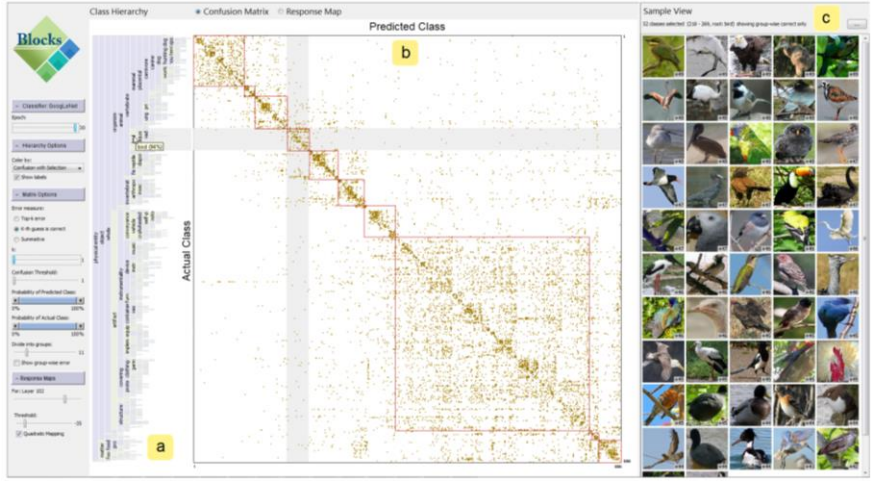


Xilinx Versal

Explicability

- Explainability is at the heart of transparency issues in AI algorithms.
- Some AI systems achieve excellent results, but the way they do it is still very opaque. They are referred to as "black boxes" because we can measure what goes in and what comes out of the machine, but not the mechanism in between.
- "Once the neural network has learned to recognize something, a developer cannot see how it succeeded. It's like the brain: you can't cut off the head and see how it works."
- For several decades, the issue of "black boxes" was not seen as crucial because "machine learning" was primarily a laboratory subject. This is no longer the case today.
- Being able to explain what is happening is crucial for multiple uses, whether it is credit attribution, medical diagnosis, or autonomous driving.
- In the absence of being able to explain the "reasoning" of deep learning algorithms, another approach is to make visible how they work.

It is not a major problem. It is very satisfying to have an explanation, and it reassures humans if an artificial intelligence system produces an explanation. But in the end, what we want is above all good reliability.“ Yann LeCun



Energy impact

Training AI models is an energy-intensive process, especially during the training phase, which can last several days.

Several factors will influence the energy cost of the process:

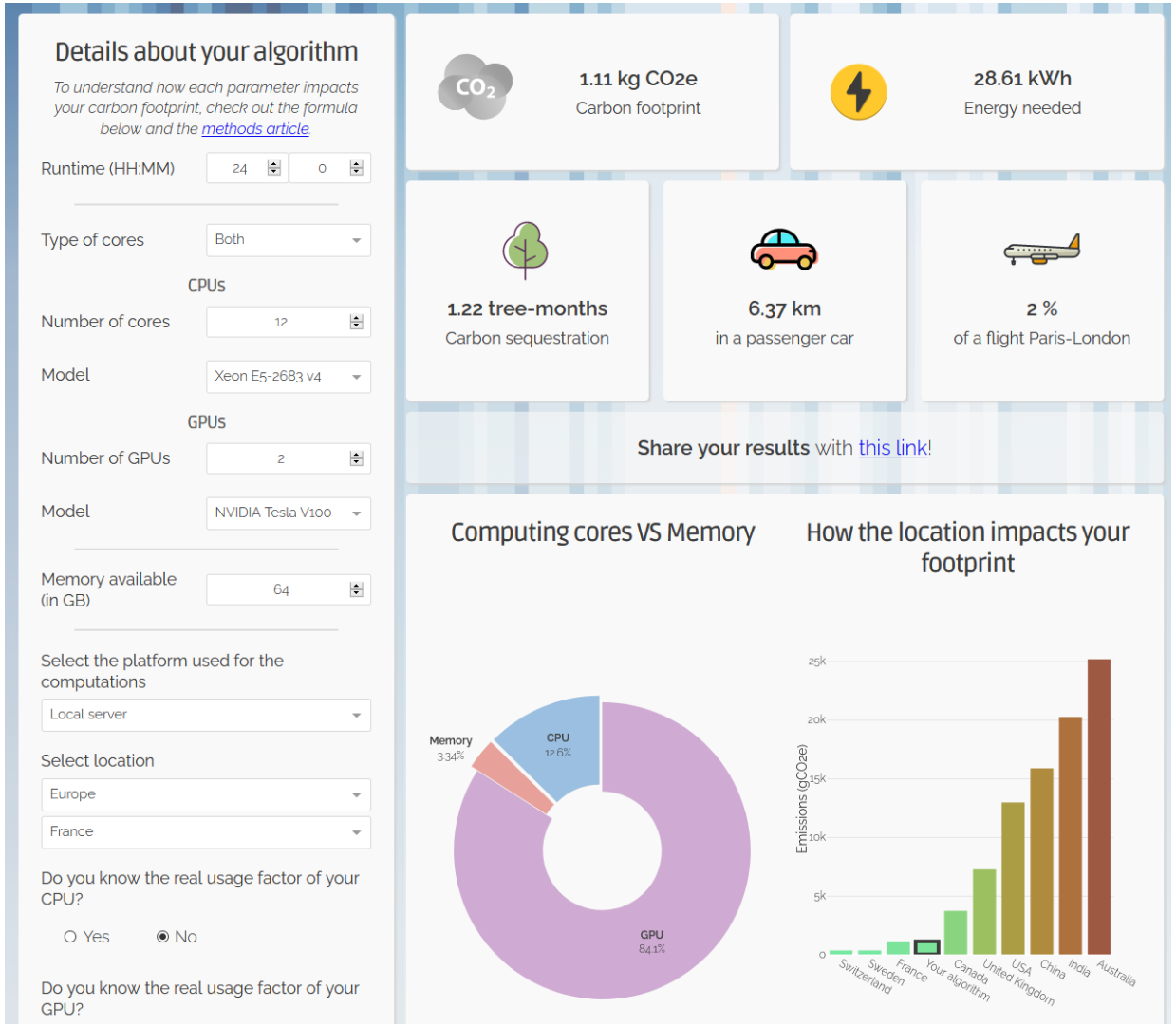
- The size of the data: for a deep network to work well, it requires a vast amount of labeled data as a learning base (for image processing, this is often counted in millions).
- The network architecture: the more complex the network architecture, the longer it will take to train.
- The type of task: the higher the level of the task, the longer the process will take.
- Optimization parameter decisions: It is not easy to find a stopping criterion, so it is common to let a program run for several hours/days to see if the system improves.

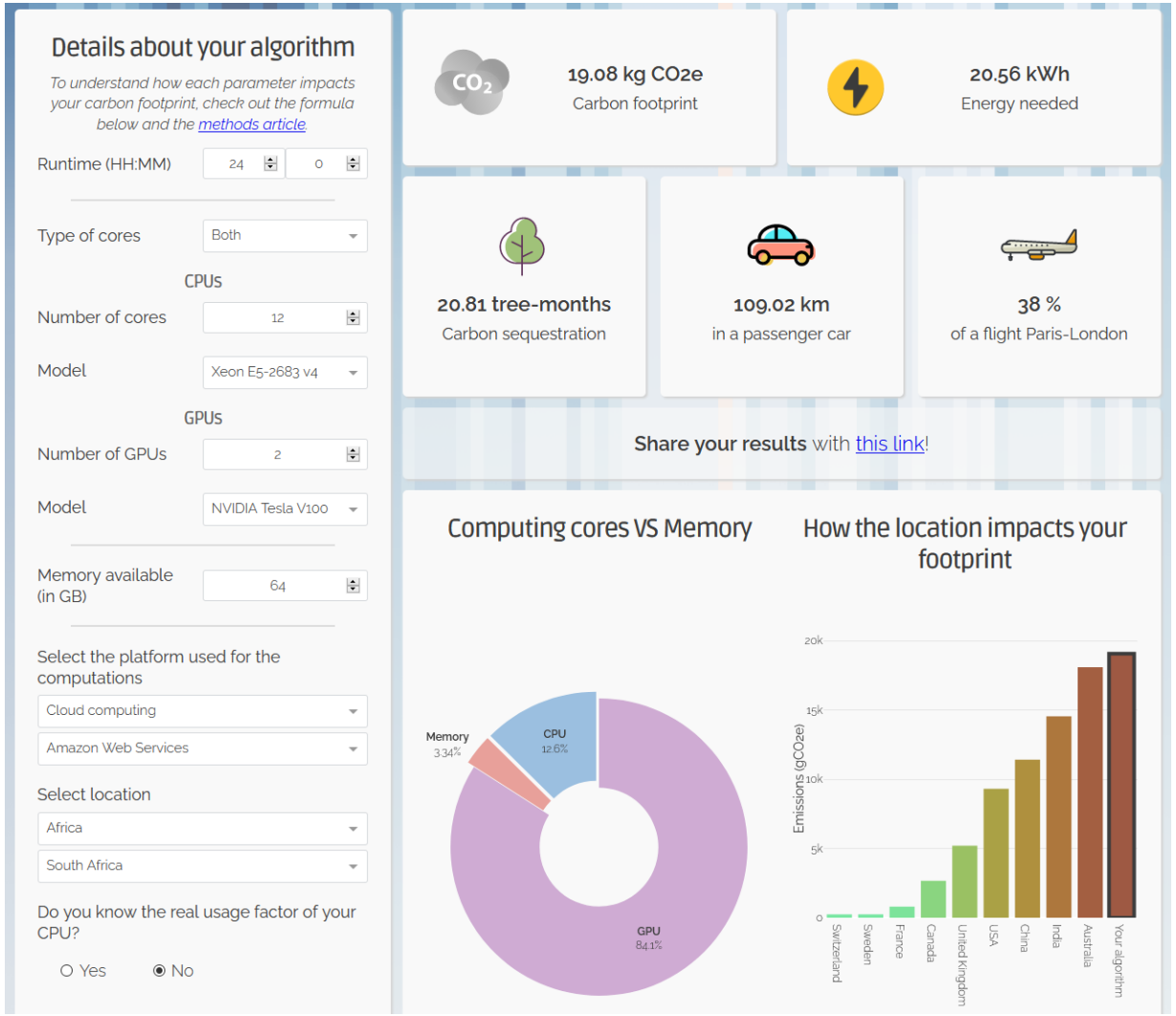
The training phases require:

- Calculation: many dedicated processors (GPUs).
- Data: data centers that store the learning bases.
- Communication: network infrastructures to route data to computing centers.

Energy impact – Cloud ?

AWS depuis Afrique du Sud

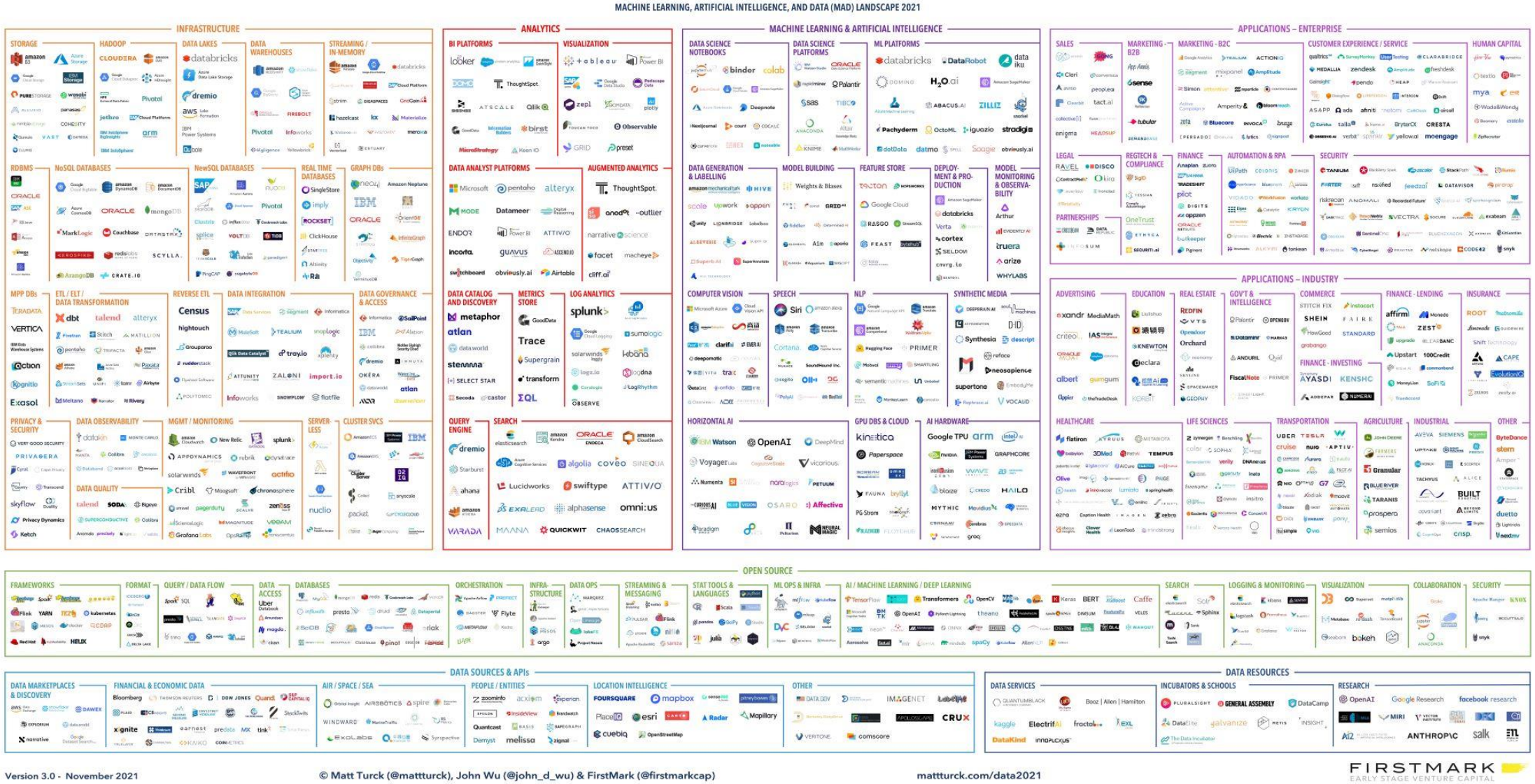




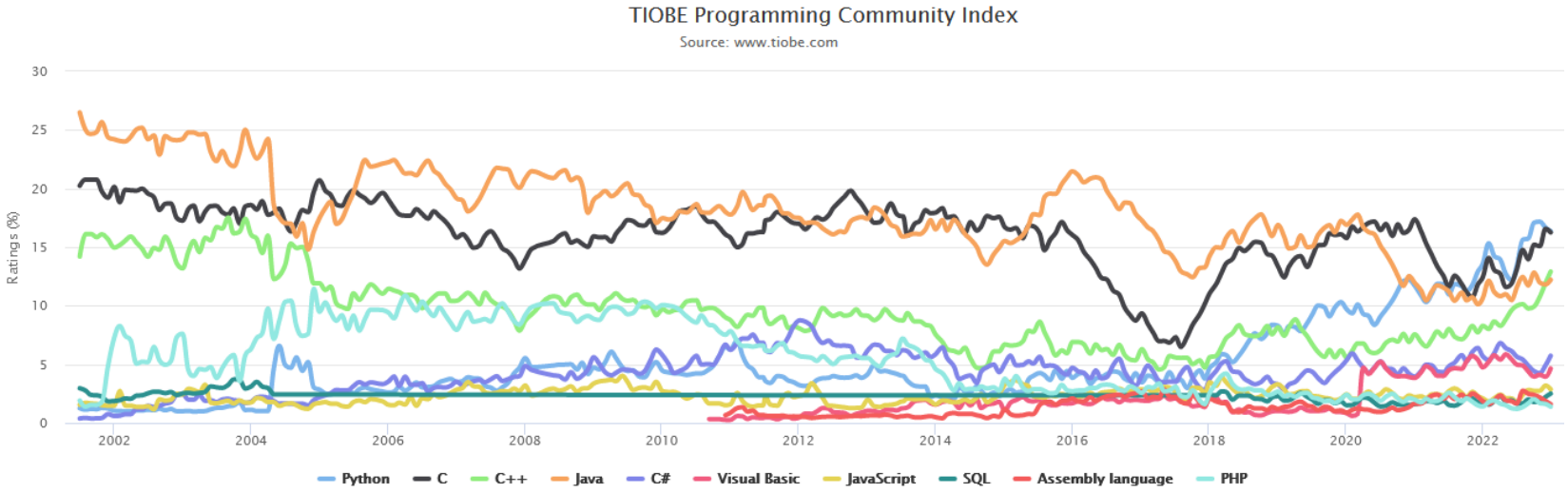
HPC CNES

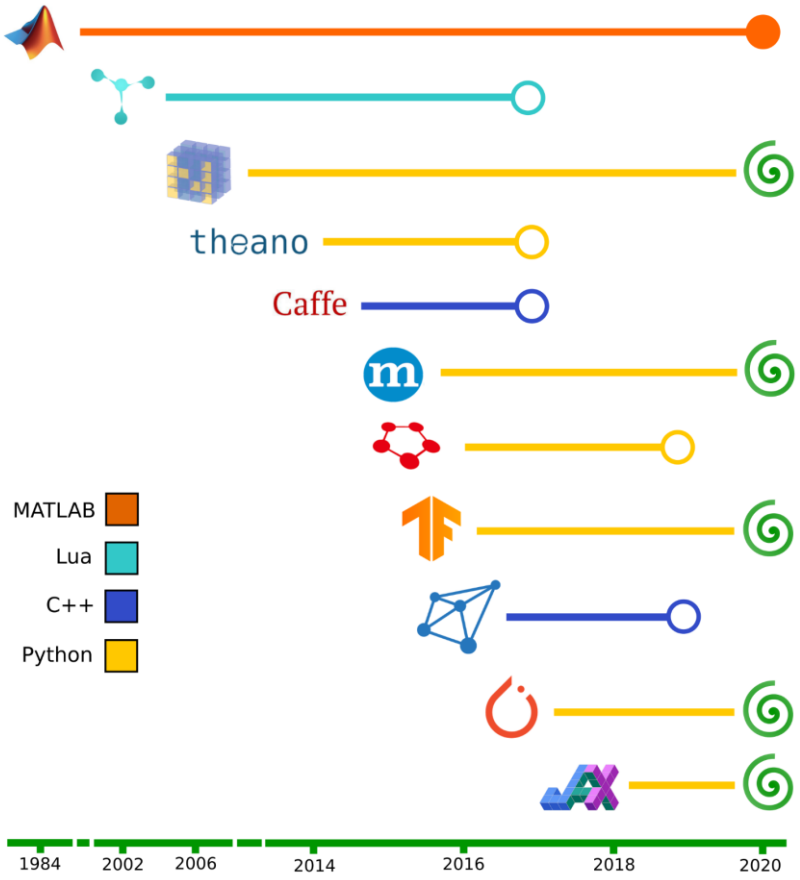
http://www.green-algorithms.org/

Comment faire de l'IA ?



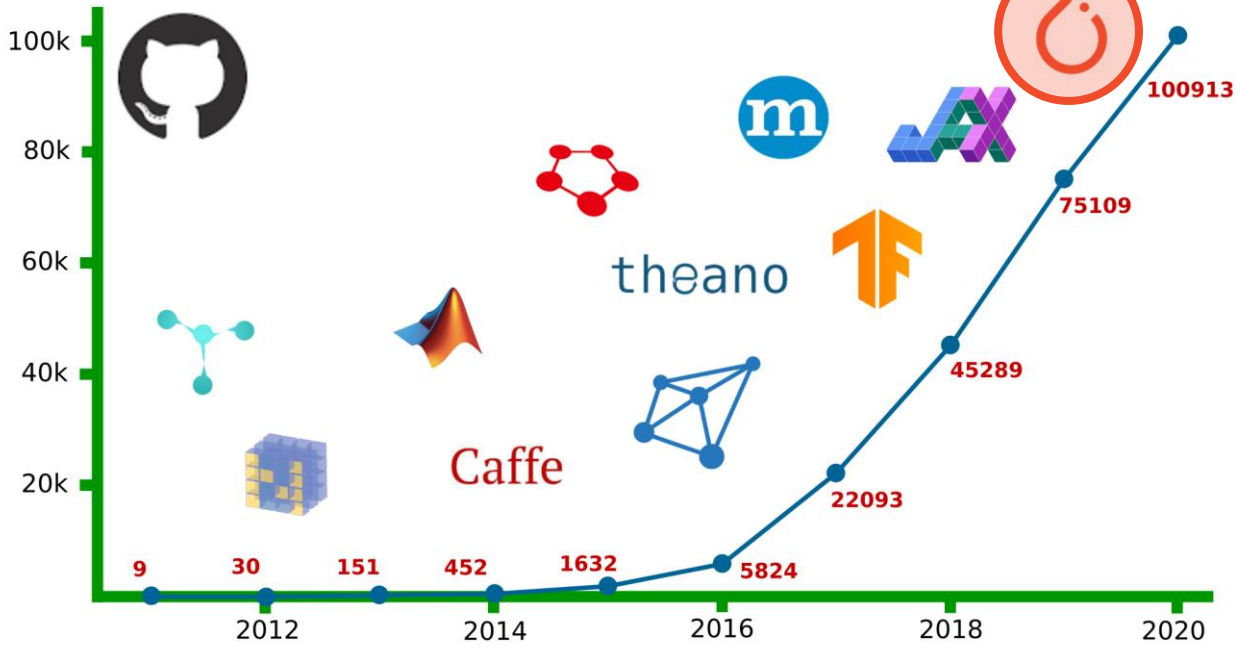
Which language ?





Which framework ?

Deep Learning



PyTorch Lightning

Machine learning

scikit learn