Machine learning, data science and artificial intelligence

Clément Levallois

2019-05-09

Table of Contents

1. Explaining machine learning in simple terms

a. A comparison with classic statistics

We will compare machine learning to a classic example from statistics: computing a regression line to identify a trend in a scatter plot.

To illustrate, we take some data about marketing budgets and sales figures in the corresponding period:
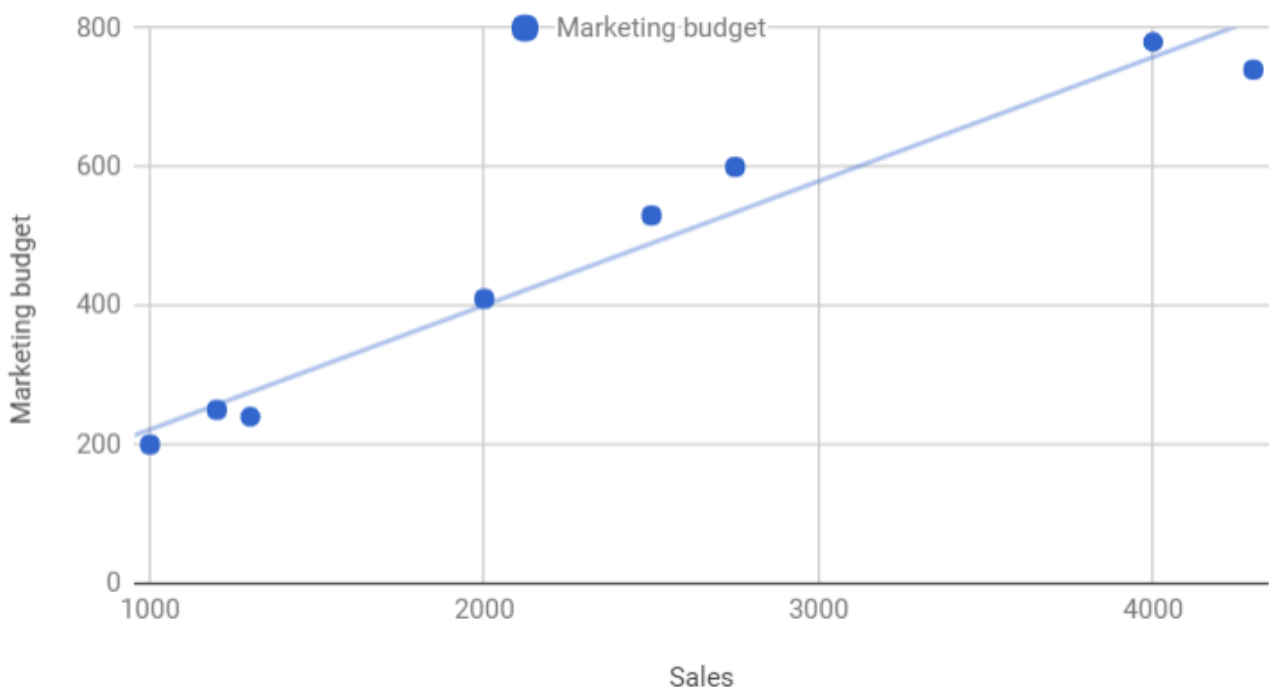

Figure 1. A linear regression

"Regular statistics" enables, among other things:

1. to find the numerical relation between the 2 series, based on a pre-established formal model (eg, ordinary least squares).

→ we see that sales are correlated with marketing spendings. It is likely that more marketing spending causes more sales.

2. to predict, based on this model:

→ by tracing the line further (using the formal model), we can predict the effect of more marketing spending

"Regular statistics" is advanced by scientists who:

1. are highly skilled in mathematics

→ their goal is to find the exact mathematical expression defining the situation at hand, under rigorous conditions

→ a key approach is **inference**: by defining a **sample of the data** of just the correct size, we can reach conclusions which are valid for the entire dataset.

2. have no training in computer science / software engineering

→ they neglect how hard it can be to to run their models on computers, in terms of calculations to perform.

→ since they focus on **sampling** the data, they are not concerned with handling entire datasets with related IT issues.

Machine learning does similar things to statistics, but in a slightly different way:

- there is an emphasis on getting the prediction right, not caring for identifying the underlying mathematical model
- the prediction needs to be achievable in the time available, with the computing resources available
- the data of interest is in a format / in a volume which is not commonly handled by regular statistics package (eg: images, observations with hundreds of features)

Machine learning is advanced by scientists who are typically:

1. highly skilled in statistics (the "classic" statistics we have seen above)

2. with a training or experience in computer science, familiar with working with unstructured data / big data

3. working in environments (industry, military, ...) where the operational aspects of the problem are key determinants (unstructured data, limits on computing resources)

Machine learning puts a premium on techniques which are "computationally adequate":

- which need the minimum / the simplest algebraic operations to run: the best technique is worthless if it's too long or expensive to compute.

- which can be run in such a way that multiple computers work in parallel (simultaneously) to solve it.

(footnote: so machine learning, in my opinion, shares the spirit of "getting things done" as was operations research in the early days)

The pursuit of improved models in traditional statistics is not immune to the notion of computational efficiency - it does count as a desirable property - but in machine learning this is largely a pre-requisite.

b. An illustration: the case of the GPU

A key illustration of the difference between statistics and machine learning can be provided with the use of **graphic cards**.

Graphic cards (or GPUs: graphics processing units) are these electronic boards full of chips found inside a computer, which are used for the display of images and videos on computer screens:

Figure 2. A graphic card sold by NVidia - a leading manufacturer

In the 1990s, video gaming developed a lot from arcades to desktop computers. Game developers created computer games showing more and more complex scenes and animations. (see an evolution of graphics, and advanced graphics games in 2017).

These video games need powerful video cards (aka GPUs) to render complex scenes in full details - with calculations on light effects and animations **made in real time**.

This pushed for the development of ever more powerful **GPUs**. Their characteristics is that they can compute simple operations to change pixel colors, **for each of the millions of pixels of the screen**

in parallel, so that the next frame of the picture can be rendered in milliseconds.

Millions of simple operations run in parallel for the price of a GPU (a couple of hundreds of dollars), not the price of dozens of computers running in parallel (can be dozens of thousands of dollars)? This is interesting for computations on big data!

If a statistical problem for prediction can be broken down into simple operations which can be run on a GPU, then a large dataset can be analyzed in seconds or minutes on a laptop, instead of cluster of computers.

To illustrate the difference in speed between a mathematical operation run without / with a **GPU**:

▶ https://www.youtube.com/watch?v=-P28LKWTzrI *(YouTube video)*

The issue is: to use a GPU for calculations, you need to conceptualize the problem at hand as one that can be:

- broken into a very large series
- of very simple operations (basically, sums or multiplications, nothing complex like square roots or polynomials)
- which can run independently from each other.

Machine learning typically pays attention to this dimension of the problem right from the design phase of models and techniques, where statistics would typically not consider the issue, or only downstream: not at the design phase but at the implementation phase.

Now that we have seen how statistics and machine learning differ in their approach, we still need to understand how does machine learning get good results, if it does not rely on modelling / sampling the data like statistics does?

Machine learning can be categorized in 3 families of tricks:

2. Three families of machine learning

a. The unsupervised learning approach

Unsupervised learning designates the methods which take a fresh dataset and find interesting patterns in it, **without inferring from previous, similar datasets**.

How does supervised learning work? Let's take an example. In a wedding reception, how to sit people with similar interests at the same tables?

The set up:

- a list of 100 guests, and 3 tastes you know they have for each of them
- 10 tables with 10 sits each.

- a measure of similarity between 2 guests: 2 guests have similarity of 0% if they share 0 tastes, 33% if they share 1 taste, 66% with 2 tastes in common, 100% with three matching interests.

- a measure of similarity at the level of a table: the sum of similarities between all pairs of guests at the table (45 pairs possible for a table of 10).

A possible solution using an unsupervised approach:

- on a computer, assign randomly the 100 guests to the 10 tables.

- for each table:
  - measure the degree of similarity of tastes for the table
  - exchange the sit of 1 person at this table, with the sit of a person at a different table.
  - measure again the degree of similarity for the table: if it improves, keep the new sits, if not, revert to before the exchange

And repeat for all tables, many times, until no exchange of sits improves the similarity. When this stage is achieved, we say the model has "**converged**".

This approach makes it possible to identify groups of people who have common points. It is obviously very useful to organize the world around us in business, from a segmentation of customers or prospects, to a classification of products in categories for evaluation or portfolio management purposes. There is a very large field of scientific research devoted to designing better clustering techniques suiting a variety of situations. One of the most popular of these techniques remains the"k-means", and was invented in the 1950s:
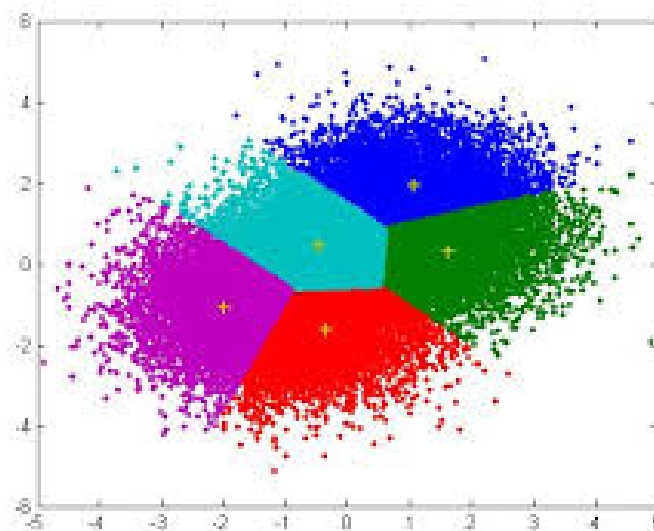

Figure 3. k-means clustering - an unsupervised learning approach

b. The supervised learning approach

Supervised learning is the approach consisting in calibrating a model based on the history of past experiences in order to guess / predict a new occurrence of the same experience.

Take 50,000 or more observations, or data points, like:

- an image of a cat, with the caption "cat"

- an image of a dog, with the caption "dog"

- another image of a cat, with the caption "cat"

etc....

- you need 50,000 observations of this kind, or more! It is called the **training set**.

- this is also called a **labelled dataset**, meaning that we have a label describing each of the observation.

> In a trained dataset, where do the labels come from?
>
> - they can be simply be provided by users of a service. For instance, pics on Instagram captioned by hashtags are exactly that: a picture with a label. The labelling is done by the users of Instagram posting the pictures and writing the hashtags below it. Instagram is a free service but the training sets it creates are of great value to the company (Instagram is owned by Facebook).
>
> - they can be produced by human workers. In practice, humans are paid a few cents per picture which they have to label (is it a cat? is it a dog? etc.). A large industry and job market is developing to perform a variety of tasks of this kind. There is a growing workforce providing their digital labor to companies in need of **data labeling** or **data curation**. See the work of Antonio Casilli for further reference.

The task is: if we give our computer a new image of a cat without a label, will it be able to guess the label "cat"?

The method:

- take a list of random coefficients (in practice, the list is a vector, or a matrix).

- for each of the 50,000 pictures of dogs and cats:
  - apply the coefficients to the picture at hand (let's say we have a dog here)
  - If the result is "dog", do nothing, it works!
  - If the result is "cat", change slightly the coefficients.
  - move to the next picture

- After looping through 50,000 pictures the parameters have hopefully adjusted and fine tuned. This was the **training of the model**.

Now, when you get new pictures (the **fresh set**), applying the trained model should output a correct prediction ("cat" or "dog").

Supervised learning is currently the most popular family of machine learning and obtains excellent results especially in image recognition, even though some cases remain hard to crack:

Figure 4. A hard test case for supervised learning

It is called **supervised** learning because the learning is very much constrained / supervised by the intensive training performed:

→ there is limited or no "unsupervised discovery" of novelty.

▶ https://www.youtube.com/watch?v=4HCE1P-m1l8 *(YouTube video)*

Important take away on the supervised approach:

- **collecting *large* datasets for training is key**. Without these data, no supervised learning.

- supervised learning is not good at analyzing situations entirely different from what is in the training set.

c. The reinforcement learning approach

To understand reinforcement learning in an intuitive sense, we can think of how animals can learn quickly by **ignoring** undesirable behavior and rewarding desirable behavior.

This is easy and takes just seconds. The following video shows B.F. Skinner, main figure in psychology in the 1950s-1970s:

▶ https://www.youtube.com/watch?v=TtfQlkGwE2U *(YouTube video)*

Footnote: how does this apply to learning in humans? On the topic of learning and decision making, I warmly recommend Foundations of Neuroeconomic Analysis by Paul Glimcher, professor of neuroscience, psychology and economics at NYU:

> 💡 this is a very hard book to read as it covers three disciplines in depth. The biological mechanisms of decision making it describes can be inspiring to design new computational approaches.
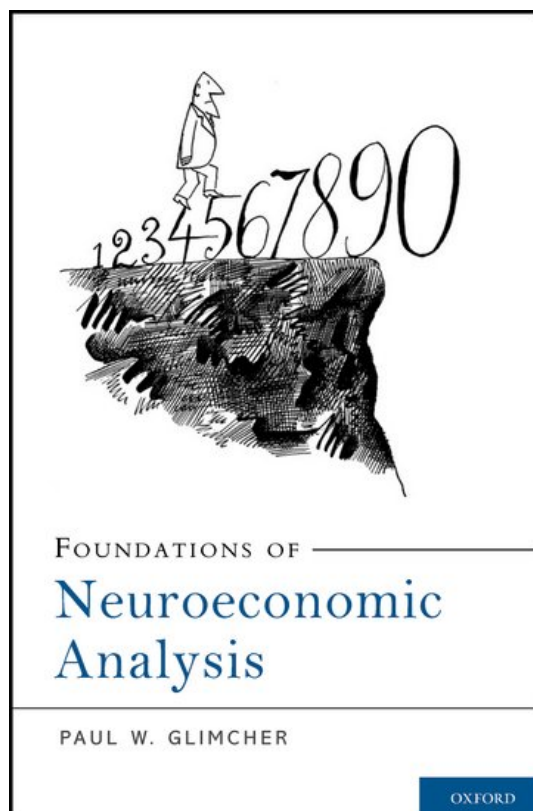

Figure 5. Foundations of Neuroeconomics by Paul Glimcher - 2010

Besides pigeons, reinforcement learning can be applied to any kind of "expert agents".

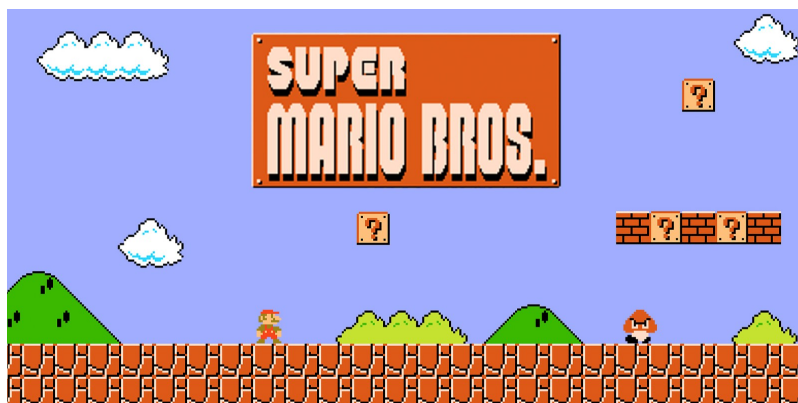Take the case of a video game like Super Mario Bros:

Figure 6. Mario Bros, a popular video game

Structure of the game / the task:

- Goal of the task: Mario should collect gold coins and complete the game by reaching the far right of the screen.

- Negative outcome to be avoided: Mario getting killed by enemies or falling in holes.

- Starting point: Mario Bros is standing at the beginning of the game, doing nothing.

- Possible actions: move right, jump, stand & do nothing, shoot ahead.

Reinforcement learning works by:

1. Making Mario do a new random action ("try something"), for example: "move right"

2. The game ends (Mario moved right, gets hit by a enemy)

3. This result is stored somewhere:
   - move right = good (progress towards the goal of the game)
   - walking close to an enemy and getting hit by it = bad

4. Game starts over (back to step 1) with a a combination of
   - continue doing actions recorded as positive
   - try something new (jump, shoot?) when close to a situation associated with a negative outcome

After looping from 1. to 4. thousands of times, Mario completes the game, without any human player:

▶ https://www.youtube.com/watch?v=qv6UVOQ0F44 *(YouTube video)*

Reinforcement learning is perceived as corresponding to an important side of human learning / human intelligence (goal oriented, "trial and error").

d. When is machine learning useful?

Using machine learning can be a waste of resource, when well known statistics could be easily applied.

Hints that "classic" statistical modeling (maybe as simple as a linear regression) should be enough:

- The dataset is not large (below 50k observations), supervised learning is not going to work
- The data is perfectly structured (tabular data)
- The data points have few features

Cases when "classic" statistics modeling is **necessary**:

- The question is about the relative contribution of independent variables to the determination of an outcome

3. Machine Learning and Data Science

Machine learning is a step in the longer chain of steps of data science.

The process was formalized as kdd: "Knowledge Discovery in Databases":
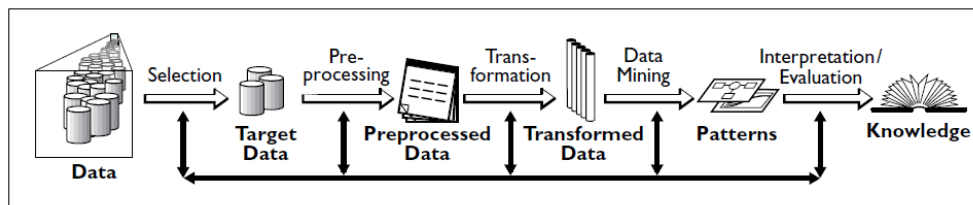

Figure 7. KDD - knowledge discovery in databases

More recent representations of the steps in data processing have been suggested, making room for the role of data visualization:

→ see the information design process by Ben Fry and this data visualization workflow by Moritz Stefaner:
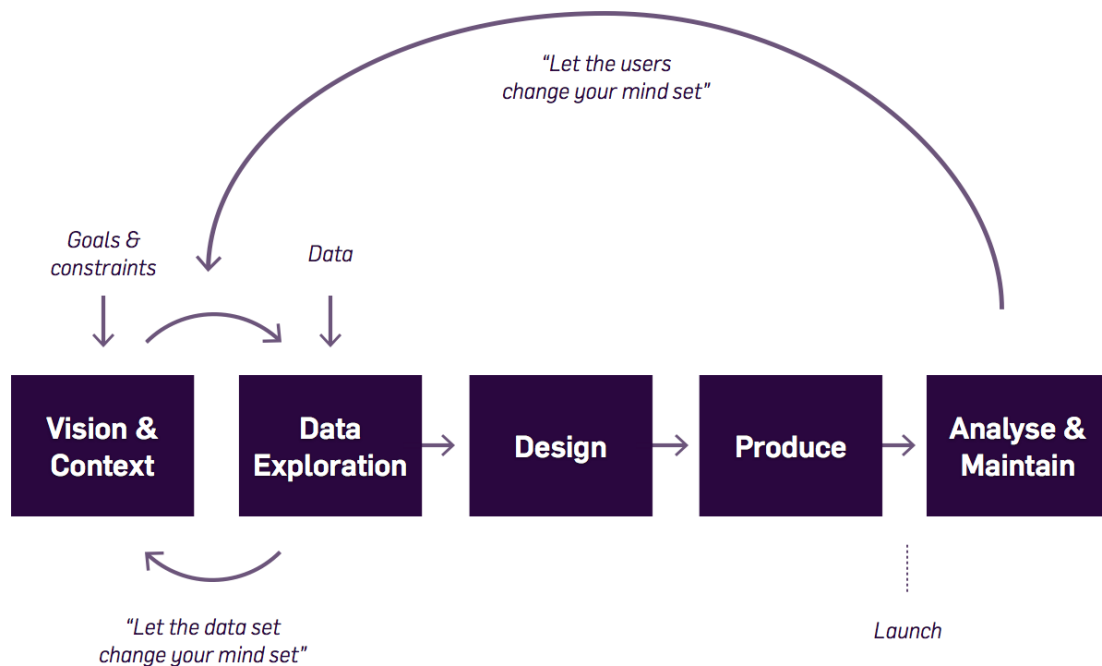

Figure 8. data visualization workflow by Moritz Stefaner

Machine learning is one of the techniques (along with traditional statistics) that intervenes at the

step of "Data mining".

What makes data scientists important is that the steps of this kdd are highly interdependent.

You need individuals or teams who are not just versed in data mining:

→ because the shape of the data at the collection stage has a huge influence on the kind of techniques, and the kind of software, that can be used to discover knowledge.

The skills of a data scientist are often represented as the meeting of three separate domains:
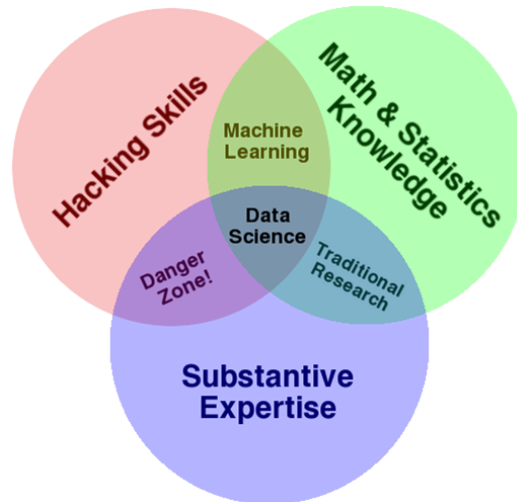

Figure 9. The Venn diagram of data science by Drew Conway

4. Artificial intelligence

a. Weak vs Strong AI

Weak AI designates computer programs able to outperform humans at complex tasks with a narrow focus (playing chess)

Weak AI is typically the result of applying expert systems or machine learning techniques seen above.

Strong AI is an intelligence that would be general in scope, able to set its own goal, and conscious of itself. Nothing is close to that yet.

So AI is a synonymous with weak AI at the moment.

b. Two videos to understand AI further

Laurent Alexandre on the social and economic stakes of **AI** (in French):

▶ https://www.youtube.com/watch?v=rJowm24piM4 *(YouTube video)*

John Launchbury, Director of DARPA's Information Innovation Office (I2O) in 2017:

▶ https://www.youtube.com/watch?v=-O01G3tSYpU *(YouTube video)*

The end

Find references for this lesson, and other lessons, here.

 This course is made by Clement Levallois.

Discover my other courses in data / tech for business: https://www.clementlevallois.net

Or get in touch via Twitter: @seinecle