Twitter Thread by Alejandro Piad Morffis



Alejandro Piad Morffis
@AlejandroPiad

This is a Twitter series on #FoundationsOfML.

■ Today, I want to start discussing the different types of Machine Learning flavors we can find.

This is a very high-level overview. In later threads, we'll dive deeper into each paradigm... ■■

Last time we talked about how Machine Learning works.

Basically, it's about having some source of experience E for solving a given task T, that allows us to find a program P which is (hopefully) optimal w.r.t. some metric M.

https://t.co/VQmL4yRVo3

> I'm starting a Twitter series on #FoundationsOfML. Today, I want to answer this simple question.
>
> \u2753 What is Machine Learning?
>
> This is my preferred way of explaining it... \U0001f447\U0001f9f5
>
> — Alejandro Piad Morffis (@AlejandroPiad) January 12, 2021

According to the nature of that experience, we can define different formulations, or flavors, of the learning process.

A useful distinction is whether we have an explicit goal or desired output, which gives rise to the definitions of 1■■ Supervised and 2■■ Unsupervised Learning ■

1■■ Supervised Learning

In this formulation, the experience E is a collection of input/output pairs, and the task T is defined as a function that produces the right output for any given input.

■ The underlying assumption is that there is some correlation (or, in general, a computable relation) between the structure of an input and its corresponding output and that it is possible to infer that function or mapping from a sufficiently large number of examples.

The output can have any structure, including a simple atomic value.

In this case, there are two special sub-problems:

■■ Classification, when the output is a category out of a finite set.
■■ Regression, when the output is a continuous value, bounded or not.

2■■ Unsupervised Learning

In this formulation, the experience E is just a collection of elements, and the task is defined as finding some hidden structure that explains those elements and/or how they relate to each other.

■ The underlying assumption is that there is some regularity in the structure of those elements which helps to explain their characteristics with a restricted amount of information, hopefully significantly less than just enumerating all elements.

Two common sub-problems are associated with where do we want to find that structure, inter- or intra-elements:

■■ Clustering, when we care about the structure relating to different elements.
■■ Dimensionality reduction, when we care about the structure internal to each element.

One of the fundamental differences between supervised and unsupervised learning problems is this:

■■ In supervised problems is easier to define an objective metric of success, but it is much harder to get data, which almost always implies a manual labeling effort.

Even though the distinction between supervised and unsupervised is kind of straightforward, it is still somewhat fuzzy, and there are other learning paradigms that don't fit neatly into these categories.

Here's a short intro to three of them ■

3■■ Reinforcement Learning

In this formulation, the experience E is not an explicit collection of data. Instead, we define an environment (a simulation of sorts) where an agent (a program) can take actions and observe their effect.

■ This paradigm is useful when we have to learn to perform a sequence of actions, and there is no obvious way to define the "correct" sequence beforehand, other than trial and error, such as training artificial players for videogames, robots, or self-driven cars.

4■■ Semi-supervised Learning

This is kind of a mixture between supervised and unsupervised learning, in which you have explicit output samples for just a few of the inputs, but you have a lot of additional inputs where you can try, at least, to learn some structure.

■ Examples are almost any supervised learning problem when we hit the point where getting additional *labeled* data (with both inputs and outputs) is too expensive, but it is easy to get lots of *unlabelled* data (just with inputs).

5■■ Self-supervised Learning

This is another paradigm that's kind of in-between supervised and unsupervised learning. Here we want to predict an explicit output, but that output is at the same time part of other inputs. So in a sense, the output is also defined implicitly.

■ A straightforward example is in language models, like BERT and GPT, where the objective is (hugely oversimplifying) to predict the n-th word in a sentence from the surrounding words, a problem for which we have lots of data (i.e., all the text on the Internet).

All of these paradigms deserve a thread of their own, perhaps even more, so stay tuned for that!

■ But before getting there, next time we'll talk a bit about the fundamental differences in the kinds of models (or program templates) we can try to train.