

CPS331 Lecture: Introduction to Learning; Fogel Discussion

last revised October 25, 2018

Objectives:

1. To understand the structure of a learning program
2. To understand why machine learning is important
3. To understand goals of learning systems (classification and prediction)
4. To understand the basic ideas of supervised, reinforcement, and unsupervised learning.

Materials:

1. Projectable of Learning Program Structure
2. Projectable of Fogel discussion question
3. Projectable of Classification Algorithm

I. Introduction

A. One of the crucial hallmarks of intelligent life is the ability to learn. In fact, the ability to learn tricks and obedience to commands is one way we distinguish more intelligent pets (such as dogs) from less intelligent ones (such as guinea pigs.)

B. Thus, enabling computer programs to learn is one crucial area that must be addressed in any attempt to produce high levels of intelligence.

1. In one sense, learning can be regarded as a component of almost any AI problem. (Cawsey points out some examples of this.)
2. In another sense, though, it is useful to study learning as an AI problem in its own right - understanding that ultimately learning becomes a part of a larger system that solves some problem.
3. Recall what Turing called “Lady Lovelace’s Objection” to the idea of an intelligent computer - a computer can only do what it is

programmed to do. In a sense, learning seeks to move us beyond that (though of course one might object that a learning computer is still doing what it is programmed to do - learn!)

4. Recall, too, Turing's idea as to how an intelligent system might be built by starting with a "child machine" and educating it in ways similar to the way a human child becomes an adult.

C. Of course, we must at some point define what we mean by "learn".

1. Tanimoto gives the following definition: "When a system learns, it improves its knowledge or its ability to perform one or more tasks. The improvement comes about as a result of information-processing activity."

- a) Note that the final phrase - "as a result of information processing activity" is intended to distinguish learning from simple programming.

- b) Example: We have already discussed how an expert system might be constructed by writing "if-then" rules. This is a form of programming. An expert system that learns, on the other hand, would infer its rules from example "cases".

- c) Indeed, some of the reasons for growing interest in machine learning have to do with difficulties with producing expert systems, including:

- (1)The "knowledge acquisition bottleneck" - the shortage of experts with sufficient domain expertise and willingness and ability to expend the time and effort needed to put that knowledge into a form it can be used by a system like an expert system.

- (2)The difficulty- if not impossibility - of formalizing knowledge as symbolic rules.

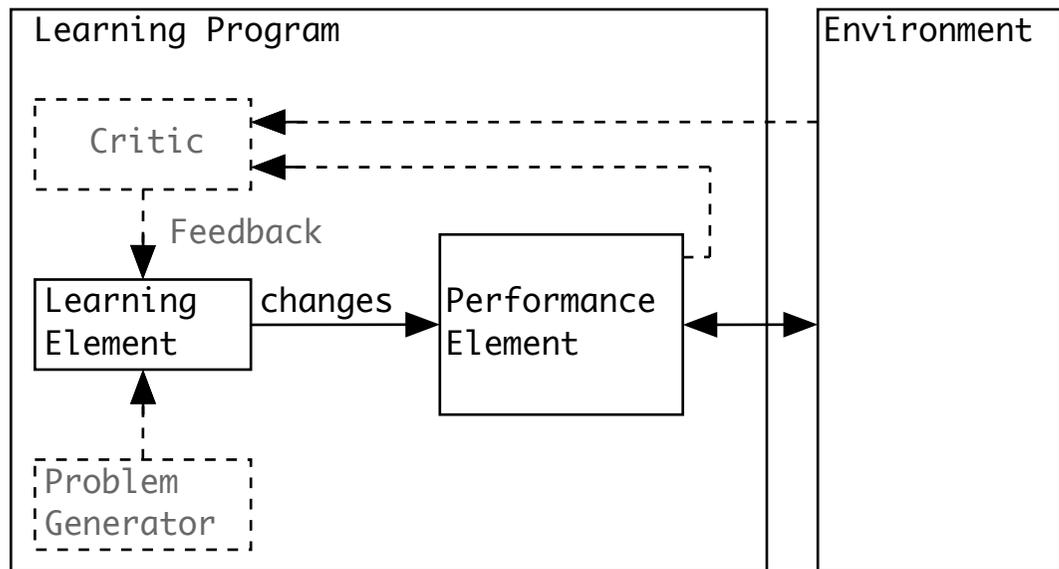
(a)Example: at the start of the course we noted that a task like recognizing pictures of different kinds of animals can do is one that even a young child can do, but one that is hard to do computationally. Try to formulate a set of predicate calculus rules for recognizing whether a picture shows a dog.

(b)We will see that some of the approaches to learning - one in particular (neural nets) yields a representation for a problem solution that is not easily capable of being formulated as symbolic rules - i.e. a human will find it hard to understand!

- 2. Russell and Norvig suggest that learning can be understood as the system acquiring the ability to compute an approximation to a function - the function that maps an input situation to the correct response.

D.In general, a learning system has the following structure:

PROJECT



1. The performance element corresponds to a program without learning. It does what the program is ultimately intended to do.
 - a) For example, if the learning program learns to play a game, then the performance element is what actually plays the game.
 - b) The performance element interacts with the environment. This may be the actual environment, or a simulated one.

For example, if the learning program learns to play a game, then the environment may be the external interface plus the opponent, or it may be another copy of the program that the learning program plays against.

2. The learning element updates the performance element.

3. The critic provides feedback on the the performance element's performance, based either on examining its output or on examining the effect of its output on the environment.

4. We will discuss the problem generator as we discuss various paradigms.

E. The one essential ingredient for many approaches to learning is data, and lots of it. Hence the interest in "big data".

1. One major source of data for machine learning is transactional data as a side effect of computer applications. For example, a company like Amazon handles over 25 million transactions per day, and for each it has a record of who bought what.

2. Learning from search queries and what result links actually get clicked helps the continual improvement of search engines like Google's.

3. Indicators of data hunger you've seen?

4. We will also see, though that some approaches to learning are not driven by data - e.g. learners that discover a near-optimal solution to a problem.

II. Discussion of Fogel Chapter

- A. In general, what was your reaction to this chapter?
- B. How is Fogel's "take" on AI different from that of Turing or Newell and Simon?
 1. According to Fogel, how has symbolic AI defined intelligence?
 2. How would Fogel define "intelligence"?
 3. What is Fogel's opinion of the Turing test and "classic" AI work such as chess-playing programs like Deep Blue? Do you agree or disagree?
- C. In Fogel's view, how should we go about trying to produce intelligent machines?
- D. Small group discussion: Do you think Fogel's critique of symbolic AI is on target? Why or why not?

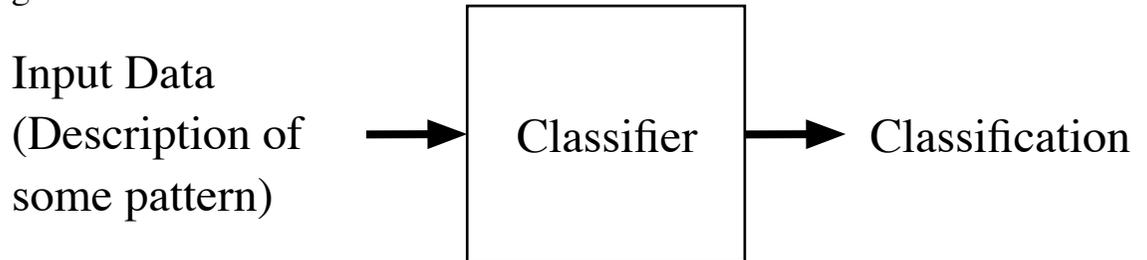
PROJECT

- E. Actually, similar views have been espoused by others as well. We'll see this, for example, in the article by Rodney Brooks we will read next.

III. Goals of Machine Learning

Learning systems can be used to accomplish a wide variety of goals.

A. Classification: A system that does classification has the following general structure.



PROJECT

Many important problems have this structure:

1. Spam filtering
2. Email routing
3. Image recognition (classify an image as to what it is an image of)
4. Facial recognition
5. Medical diagnosis
6. Optical character recognition
7. Speech recognition
8. Other

ASK

B. Closely related to classification: prediction - essentially anticipating what class a particular data point will belong to.

1. Determination of credit worthiness (classification as likely to be low-risk, moderate-risk, high-risk)
2. Evaluation of applicants for a job or graduate school (classification as likely a good fit, likely not a good fit)
3. Other? (ask)

C. Prediction of numerical values rather than simple class membership -- e.g. the likely selling price of a home (based on what similar homes have sold for).

D. Clustering: Grouping items into categories by some measures of similarity

1. Customer segmentation in marketing (supporting targeting of advertising)
2. Recommendation systems (you might also like ... - based on what others in same cluster have liked or bought)
3. Other?

ASK

E. Association (often the result of mining databases of transactional data) - e.g. Basket analysis in retail (many customers who bought X also bought Y)

F. Acquiring skills (e.g. self-driving cars)

IV. Learning is an Inductive Rather than Deductive Approach

A. Rule-based systems - including predicate calculus - are deductive systems. Given a set of rules or axioms, one can deduce new knowledge by a process such as modus ponens or a proof.

Example: Given the premises all humans are mortal and Socrates is a human, we can infer that Socrates is mortal.

B. Learning is an inductive approach - we make inferences from a set of observations, but there is always the possibility that our inferences may be inadequate - we may infer not enough or we may infer too much.

Example: If we observe that Socrates and Aristotle and Plato all died, and know that each was a Greek philosopher, should we induce that all Greek philosophers are mortal. Or should we induce instead that all Greeks are mortal (not just philosophers), or perhaps that all philosophers are mortal (not just Greek ones)? Or should we induce that all humans are mortal?

Example: If we observe 100 Gordon students to all be under 25 years old, should we induce that all Gordon students are under 25?

V. Learning can occur in several ways.

A. One approach is known as supervised learning. This approach is often used with classification problems

1. The classification may be as simple as “instance of the concept or not an instance of the concept”, or it may involve multiple categories.
2. There is a set of training data, which consists of pairs of the form (input data, classification), often drawn from data about past observations. (This kind of data is called labeled data)

- a) The problem generator presents the pairs to the performance element one by one.
 - b) The critic compares the classification produced by the performance element with the correct classification from the training data.
 - (1) If the performance element gives an incorrect classification, then the learning element modifies the performance element to be more likely to give the correct answer for this data if it is tried again. (Note that I said "more likely" - it turns out that learning is generally better when done in small steps rather than all at once.)
 - (2) The system is expected to learn how to correctly classify each training pattern. The learning process can stop when this point is reached.
 - (3) An essential aim of this sort of learning is generalization: the ability of the system to correctly classify data not part of its original training data.
3. Under this approach, there is a “teacher” who chooses the training data to be presented to the system. However, this differs from conventional program maintenance in that the teacher does not directly modify the knowledge base; rather, the learning element of the program modifies the performance element in response to how it handles the data presented to it.

Example: A junk mail filter is typically trained as follows

- (1) If the user determines that an incoming piece of mail is junk, the user indicates this in some fashion (e.g. by clicking a button). This causes the filter to adjust its classification rules so that similar mail in the future will tend to be classified as junk.

(2) A junk mail filter generally quarantines incoming mail it classifies as junk in a special “junk mail” folder. If the user determines that a piece of mail the filter has placed in this folder is not really junk mail, he/she can indicate in some way (e.g. by clicking a different button) that this particular piece of mail is not junk. This causes the filter to adjust its classification scheme so that similar mail in the future is not likely to be classified as junk.

(3) Notice, the use of words like “tend to” and “not likely” in the above. Learning is typically incremental, so it may take human classification of many pieces of mail for the filter to “learn” the correct scheme.

4. The Danger of Overfitting

a) One insidious problem that can arise in supervised learning is called overfitting. We say that a learner has overfit when it has discovered a pattern in the training data that is unique to the training data but is not really part of the solution. We will discuss this in the context of the most common use of machine learning: classification problems.

b) The following story (though itself likely an urban legend) illustrates the problem:

In a probably apocryphal story, an AI research team tried to train a computer to detect a tank among trees in a forest. They showed it 50 tanks in a forest so it could learn what a tank looked like and showed it 50 forests with no tank so it could learn about its absence. It learned to recognize tanks and non-tanks both in the training set and in the pictures held out to serve as the testing set.

Unfortunately, though, it has not learned the right thing. It turned out that the pictures of tanks used to train it and test it were darker than the pictures that did not contain tanks. What it

actually learned is the differences in darkness/lightness of the images since many of the pictures with a tank were taken on cloudy days, and the ones without a tank on sunny days!

c) There are at least two general strategies used to reduce this danger.

(1) One can take the set of data one has for learning and divide it into two parts. One part, called the training data, is used as input to the training algorithm. The other part, called the test data, is held out until the algorithm has been trained. Then, the trained algorithm is run against the test data. If it does not handle this data well, it is likely that it overfitted the training data.

(2) A second strategy is to stop short of perfection during training - i.e. the algorithm is considered trained when it correctly classifies most of the training data, but not all of it. This can be done by strategies such as penalizing learned rules based on complexity.

5. Can you think of examples where humans learn using supervised learning?

ASK

Example: we might use this approach to teach a child to recognize different kinds of animals. The teacher shows the child pictures of various animals and tells the child what each is. The hope is that the child will learn to recognize other similar animals when shown a picture.

B. A second approach is known as reinforcement learning

1. The system performs either in its intended environment or a simulated environment, with the only feedback being whether or not its performance is correct. (The system is not given the correct answer if it did the wrong thing).

For example, if the learning program plays a game, the feedback may be simply whether or not the program won the game.

- a) The feedback is sometimes referred to as a “reward” - with a positive reward if the program did the right thing, and a negative reward if it did not.
 - b) Whenever the critic produces a negative reward, the learning element updates the performance element appropriately. The expectation is that the program will eventually learn to make the correct choice when confronted with similar situations in the future.
2. There may or may not need to be a problem generator, per se, depending on whether the system is designed to learn in a real environment or a simulated one.
3. Again, can you think of examples where humans or animals learn this way?

ASK

- a) Example: we learn a lot of skills this way ourselves: riding a bicycle, shooting a basketball through the hoop, playing a particular game ...
- b) Example: training a dog [reinforcement is either a treat or a scolding].
- c) Example: if you’ve ever watched a squirrel trying to get into a bird feeder, you know that many animals learn this way too!

C. A third approach is known as unsupervised learning.

1. Rather than being presented with right or wrong answers, the learning system extracts patterns from data, or conducts some sort of “thought experiment.”
2. In this approach, there really is no problem generator or critic per se.
3. This approach is used for clustering and learning associations.
4. Among humans, the ability to learn in this way distinguishes an expert in a field from a neophyte. This approach to learning is exemplified by the theses or projects required in many undergraduate or graduate programs.
5. However, this kind of learning is not limited to human experts. In fact, for a new-born child this is the first kind of learning the child does, as he/she gradually discovers his/her bodily faculties and how to control them! For example, initially the sounds young children make are meaningless babble, but as the child begins to learn how to speak he or she begins to focus on making the sounds that the child hears others in its environment making - that is, the phonemes of the language the child is about to learn.

D. Learning systems come in several rather distinct “flavors”. The following classification comes from *The Master Algorithm* by Pedro Domingos.

1. Symbolic learning - in which the system learns rules, or parameters of rules.
2. Evolutionary/Genetic learning - a very different approach, modeled on natural selection.
3. Connectionist learning - in which the system learns connection weights in a neural network.

4. Bayesian learning - in which the system learns probabilistic models based on observed data.
 5. Resemblance learning - in which systems learn from similarities between yet-to-be classified and already-classified entities.
 6. In the next lecture, we will focus on symbolic learning approaches. We will discuss genetic, connectionist, probabilistic and resemblance based learning in subsequent lectures.
 7. Later, we will focus on some more novel approaches inspired by biological systems that cannot perhaps be considered learning in the strictest sense, but are nonetheless interesting.
- E. Because human learning, in general, is not well understood, it should not be surprising that machine learning is still very much a research frontier, with very different approaches used to solve different problems. The literature on this topic is extensive to say the least!