MACHINE LEARNING :
The Necessity of Order (is order in order ?)

A. Cornuéjols

Laboratoire de Recherche en Informatique (L.R.I.)
Bât 490, Université Paris Sud
91405 Orsay Cedex (France)
(antoine@lri.fr)

In myriad of human-tailored activities, whether in the classroom or listening to a story, human learners receive selected pieces of information, presented in a chosen order and pace. This is what it takes to facilitate learning. Yet, when machine learners exhibited sequencing effects, showing that some data sampling, ordering and tempo are better than others, it almost came as a surprise. Seemingly simple questions had suddenly to be thought anew : what are good training data? How to select them? How to present them? Why is it that there are sequencing effects? How to measure them? Should we try to avoid them or take advantage of them?

This chapter is intended to present ideas and directions of research that are currently studied in the machine learning field to answer these questions and others. As any other science, machine learning strives to develop models that stress fundamental aspects of the phenomenon under study. The basic concepts and models developed in machine learning are presented here, as well as some of the findings that may have significance and counterparts in related disciplines interested in learning and education.

1. Introduction

Imagine this new Kibur's nightmare. The game that says : "Order an experience and experience order". How deceptively simple it looks. You just have a box with a transparent top, a landscape like rubber surface inside, and a ball rolling on this surface. And a few knobs on the side. Now, you place your box on a horizontal area. You are told that after you have turned all the knobs on the side, in whatever order suits you, the ball has to be located just under a mark on the top of the box. Each knob, when turned, has an effect on the rubber surface and modifies its shape, rotating it along some axis or accentuating wells and mountains, and the rolling ball, in consequence, changes its position. Once every knob has been turned, it is known that the lowest point on the rubber surface is located below the mark on the top side of the box. The trick is that the ball, which only descends in the local well, will not necessarily be found in the lowest spot of the surface. Only by carefully selecting the order of the knobs' turning, will the ball be guided to its desired place. How should you go then to select one of the very many (n! if there are n knobs) possible orderings? You play a little bit, get older if not order, and start to notice patterns. There is, for instance, this knob which can be turned anytime without affecting the final position of the ball, or the other that must be turned third if success is to be obtained. By carefully selecting experiments, you learn the laws of your Kibur's box. Or so you believe.

Now comes the time when you are ripe for variants. In one of these, you must manage to place your ball under other selected spots on the transparent top. This time you may choose whatever knobs suits you without having to turn them all. That seems easier? Go and try it. Then, there is this ultimate experience. You change the ball, and place a new one inside the box, with a kind a Velcro like covering that makes it slow in its changes of location on the slopes of the surface. Now, you are ready to experiment with out of equilibrium control. You can play with the speed of the surface transformations to guide your ball. A world full of new possibilities and challenges has opened up. You now have to control the choice of the knobs to turn, the order in which they are turned, and even the intervals between each turn. Experience order, or ordeal experience? You begin to wonder.

In many learning situations, the sequence of training experiences is a key parameter to learning. In education, for instance, teachers pay a great deal of attention to the teaching schedule. In particular, they will carefully select the order of the training exercises. So will the students themselves when preparing for an exam. Clearly, some choices and orders of exercises are better than others, while the delay (hours, days, or weeks) between problem solving sessions is also of importance. This is not true only of human learning and concept acquisition. It has been found that rats, taught to press a lever to obtain food, learn quicker when rewarded first for partial performance of the desired response, for instance, simply for facing the end of the cage in which the lever sits, then for increasingly more precise behavior.

In fact, some artificial learning systems that are coming of age now are equally sensitive to the sequencing of data presentation. The same questions abound for machine learners as for the natural organisms. What kind of sequencing effects show up in learning? Why are some learners (artificial or natural) affected while others are not? Are there general rules to choose the (most) appropriate teaching sequence given one learner and a goal state? These questions are difficult ones. The hope for the scientists involved in this research is that the more easily controllable machine learning setting will provide results that will hold for large classes of learning systems including natural ones, providing levers for further investigations in the natural learning area.

This chapter is intended to give a flavour of the research directions and the state of the art in machine learning regarding these questions. However, because the approach followed in machine learning is specific and original, it is interesting to sketch first a general perspective, one which may seem somewhat abstract and distant from natural and artificial learning systems, but which highlights crucial questions and essential parameters that play a role in sequential learning. This sketch will put in perspective the approaches and findings in machine learning and its links with the corresponding issues in psychology and in the educational sciences. We thus kindly ask the reader to spend some time examining these issues in a general and abstract setting, and in a sense to go back to our Kibur's game.

2. What does machine learning say on learning

Machine learning is a very young discipline compared to other scientific ones, say physics or chemistry (see for instance Mitchell (1997), Russell & Norvig, (2003) or Cornuejols & Miclet (2002) for the French speakers). The first computer models incorporating learning or adaptive capacities go back to the 1950s with works on cybernetic turtles and on automatic checker players, and it was not until the 1980s that a significant number of researchers dedicated themselves to this discipline. Yet, if one looks on the Internet to examine the content of the courses offered all around the world on machine learning, one will find that there is a large core of shared material, and, more importantly, that there seems to be a total consensus on what is learning and what constitutes the main lessons to be remembered about machine learning. This is quite extraordinary given the sheer

complexity of the subject matter: learning. Has machine learning found what has eluded philosophy, psychology and neurobiology for centuries? In this perspective, let us to examine the course taken by machine learning over the last half century.

The first attempts at programming learning systems were oriented towards adaptive behaviours, either in cybernetic turtles that learned to exit a labyrinth or in the adaptation to adversaries in the checker game. The "learner" was subject to reinforcement signals at times, positive or negative, and the learner's task was to maximize a kind of cumulated gain over time. These earlier systems were aimed at demonstrating the possibility of machine adaptation. They implemented clever heuristics while almost no theory about adaptive systems existed yet. In addition, they were following in cybernetics's, with little or no concern for "knowledge" and reasoning. The 1960s witnessed the development of the, then new, field of "pattern recognition" where the problem is to find ways of recognizing instances of types of objects by observing simple characteristics, or features, of these instances. A theoretical account emerged, mostly of a statistical nature, and new algorithms were developed. Among them the illustrious perceptron (see Rosenblatt (1962), Minsky & Papert (1988)), ancestor of the now so commonly used formal neural networks. We will return to this approach to learning, because, in more than one way, the current point of view is only a continuation of it.

Towards the end of the 1960s, the pattern recognition approach to learning was overshadowed by a new interest in cognitive aspects. This transition was inspired by the dominant artificial intelligence tradition that put emphasis on knowledge representation, rule-based inferencing, and more generally symbolic manipulations as the basis of intellectual activity. The work on machine learning changed in nature during that period which extends to the earlier 1980s. PhD theses, which represent the common quantum of activity in science, tended to aim at producing computer models of cognitive activities measured in human subjects, such as analogy making, learning the basis of geometry or making conjectures in algebra. The hypotheses tested in these works were related to the division between working memory and semantic memory, the role of procedural versus declarative knowledge, the structuring of knowledge in memory, and so on. In line with cognitive science, but in contrast with the current dominant view in machine learning, learning was considered as an on-going activity. In addition, because the learner was supposed to learn something from a teacher that was cognitively equivalent, learning was considered successful to the extent that the learner correctly identified the concepts that were in the head of the teacher. This period was in many ways the golden age of the collaboration between cognitive science and machine learning. It didn't last for two reasons at least. First, the cognitive architectures developed in these computer programs, impressive as they were, depended too much on ad hoc tunings of the very many parameters hidden in the system. It was difficult to extract general guiding principles from these experiments, let alone a general theory of learning. There was therefore uncertainty and doubts about the very nature of the scientific investigation associated with machine learning. Second, just at this time appeared the parallel distributed processing revolution also called the neo-connectionist period. With it, everything was changed. It brought the pattern recognition paradigm back in machine learning, with the idea that learning is essentially the search for a good approximation of some hidden concept known from a training sample. Also, because neural networks are adaptive systems based on differential equations and on optimization techniques, this neo-connectionist revolution attracted theoretically oriented computer scientists to the field and as a consequence put more and more emphasis on convergence problems, and on complexity issues. More than twenty years later, we are still mostly in the wake of this turning point.

According to this simplified perspective, learning chiefly comes in three guises: supervised learning, nonsupervised learning, and reinforcement learning. Since supervised learning has received most of the attention and has shaped the whole approach to learning, we start by describing the main ideas behind it.

The task in *supervised learning* can be defined in simple terms: a learning agent receives training data in the form of pairs $(\boldsymbol{x}_i, y_i)$ where $\boldsymbol{x}_i$ is an example described by a set of simple characteristics, often called features, and y_i is its associated output value that can be of various forms depending on the learning task. For instance, examples could be descriptions of objects or situations that are to be assigned to one of a predefined set of classes, as in an optical character recognizer that must output the name of the character ('A' through 'Z') corresponding to the input digitalized image. This is accordingly called a *classification task*. In the case of a Boolean output, one speaks of a *concept learning* task since the system is supposed to recognize instances of a concept against instances of anything else. If the output is taken from a continuous range of values, as is the case in the prediction of a trade market index, for instance, then one speaks of a *regression task*. In all cases, the output value is supposed to be computed from the corresponding input according to a hidden function (possibly corrupted by noise) or a hidden dependency, as is the case for a probabilistic dependency. The task of the learner is, from a limited collection of training data, to discover this dependency, or at least to find a good approximation of it.

In order to be well-posed, this problem needs additional constraints. First, the hidden or target dependency must be fixed a priori, or at least changing sufficiently slowly that regularities can be reliably identified in the data. Second, except for very specific protocols where the data are supposed to be complete in some sense, it is unrealistic to demand perfect identification of the target dependency from the learner. A good approximation is enough. But what does that mean? In supervised learning, it is assumed that the ultimate goal of learning is that the system makes predictions as accurate as possible on future events that can happen in the same universe as the one in which the learner was trained. This requirement involves two parts. The first one measures the goodness of fit of the agent's prediction compared to the actual output for a given event. This "loss function" can, for instance, be a 0-1 function where it counts 0 if the prediction was correct, and 1 otherwise. One can then get the number of mistakes on a given set of test data. It can also be the squared difference of the prediction and of the correct output, or it can take the form of a loss matrix that counts differently errors of different types. For instance, as in medicine, a false negative could be more costly than a false positive. The second part allows one to compute the expected cost of using the agent's knowledge in the universe which includes unseen events. It takes the form of a probability density function over the events in the universe.

Thus, supervised learning can be seen as the search for a hypothesis as close as possible to the target dependency, which means that its expected cost for, as yet, unseen events will be minimal. The theoretical analysis of supervised learning therefore involves three essential interrogations:

- First, given that one cannot compute directly the expected cost associated with a candidate hypothesis that one seeks to minimize, what should be the optimizing criterion defined over the available information?
- Second, in which space should one look for candidate hypotheses? Are there conditions on this hypothesis space, or is any space appropriate?
- Third, how the exploration of the hypothesis space should be organized in order to find an hypothesis optimizing the criteria of interest?

We shall treat these three questions in turn.

Several evaluation schemes over hypotheses have been proposed for inductive purpose, they all revolve around three main "*inductive criteria*" that correspond to seemingly intuitive and reasonable inductive principles:

- The first one says that *the best hypothesis is the one that most closely fits the data.* The underlying reason being that an hypothesis that fits well the known data should also fit the unknowns. This principle is called the empirical risk minimization principle, where the empirical risk measures the misfit, also called the loss, on the known data.
- The second one favours *the hypotheses that are the most likely given the data*. This requires to estimate both a prior over the hypotheses and conditional probabilities. The maximum likelihood principle is one well-known variant. It states that if all the hypotheses are equally likely before the observation of the training data, then one should choose the most likely once the training data has been accounted for. In other words, one should choose the hypothesis that can more easily explain the data.
- The third one is inspired from information and communication theories and states that the best hypothesis is the one that allows one to transmit with as few bits as possible the information contained in the training data. In essence, it is closely related to the intuitive notion that *the best explanation of a set of phenomena is the simplest one*.

Depending on one's background, inclination, and requirements from the problem under study, a researcher will adopt one or the other of these inductive principles. Fortunately, there are strong links between them, and the choice of one particular principle does not usually have a large impact on the result.

Now that potential evaluation criteria have been defined over the candidate hypotheses, how should we choose the hypothesis space over which the evaluation will take place in order to select the best, or, at least, a promising hypothesis? Apart from feasibility concerns, why shouldn't we choose the largest or richest possible hypothesis space, in the hope that this way, we have assurance that such an hypothesis space contains at least one hypothesis very close, if not identical, to the target dependency?

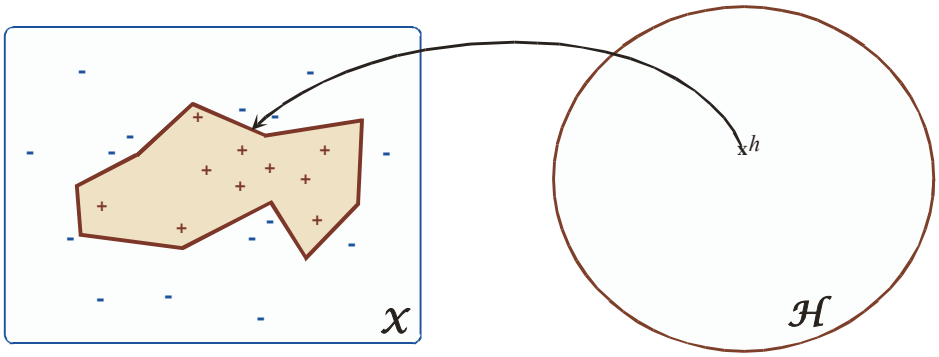

Figure 1. Inductive concept learning involves exploring an hypothesis space $\mathcal{H}$ in order to find an hypothesis h that describes a region of the example space $\mathcal{X}$ coherent with the known training instances.

The study of inductive learning in fact shows that there is a fundamental trade-off between the richness of the hypothesis space, in some sense that can be technically defined, and the required number of training data needed to compute a good hypothesis. In essence, the richer the hypothesis space, the weaker is the link between the estimated value of one hypothesis on the basis of the training data and its real value. Only more training data or insightful information from experts can tighten this link. This is why so much of the discussion and arguments in machine learning revolves around the choice of the hypothesis space or of the related knowledge representation

language. The problem, of course, is that too a constrained hypothesis space can prevent finding a good hypothesis therein. Part of the game in inductive learning is, therefore, to adaptively select the best hypothesis space. Nothing however can outperform the informed guess of experts of the domain under study.

Finally, there is the issue of the actual exploration of the hypothesis space, once chosen, in order to find a good, or even the best hypothesis according to the selected inductive criterion. The search procedure critically depends upon the structure or absence thereof that the hypothesis space can be endowed with. There are three main possibilities:

- The first one is when a relation of relative generality can be defined over the space of hypotheses. In this way, it is possible, by directly considering their expressions, to determine whether one hypothesis is more general, more specific or not comparable to another one. More importantly, it is usually possible to define generalization and specialization operators that allow one to obtain hypotheses more general or more specific than a given one. If available, this generality relation is very helpful in that it allows exploring the hypothesis space in an informed way by following meaningful directions. For instance, if a given candidate hypothesis does not cover or explain some known positive instance of the target concept, then, only a more general hypothesis is acceptable. The exploration of the hypothesis space can then be carried out by applying generalization operators to the incorrect hypothesis. In contrast, specialization operators should be used if the candidate hypothesis wrongly covers negative instances of the target concept. This is the basis of what is often called “symbolic machine learning”. The difficulty of defining a relation of generality over the hypothesis space is more than counterbalanced by the resulting much greater efficiency of the search procedure and the accompanying comprehensibility of the learning process.
- The second possibility corresponds to the case when no generality relationship can be found over the hypothesis space. For instance, the use of neural networks does not allow for such a generality relationship to be defined. In that case, one has to step back on less powerful search procedures. In effect, this means procedures that are based on gradient descent techniques. Using the chosen inductive criterion, one estimates the merit of the current candidate hypothesis and of its immediate neighbours, and follows the direction that leads to the seemingly greater improvement. The notorious back-propagation algorithms for neural networks or the genetic algorithms mechanisms are nothing other than variants of gradient-based search procedures. For their greater range of applicability, one pays for their relative inefficiency and opaque character.
- Finally, it can happen that it is not even possible to define an hypothesis space per se, distinct from the example space. Learning then does not output anymore hypotheses about the world but only decisions based on comparisons between a current input and known instances. The famous nearest neighbour technique which decides that an input is of the same class as the class of the nearest known example is an instance of this family of algorithms. This corresponds to the weakest possible type of learning. In order to give good results, it requires the knowledge of a large quantity of examples.

The framework for inductive learning that has been sketched here has similar counterparts in unsupervised learning and somewhat in reinforcement learning as well. It has provided a sound theoretical basis for induction. This, in turn, can be used to explain the properties of well-known learning algorithms that were first obtained heuristically, but it has also lead to new powerful inductive algorithms that were motivated from theoretical interrogations, such as the so-called Support Vector Machines (see Vapnik (1995), Schölkopf et al. (1999)), or

the Boosting meta learning technique (see Freund & Shapire). There are now myriads of applications for learning techniques that go under the generic name of data mining. Machine learning is thus a well-established field with seemingly a strong theoretical basis.

Hence, has machine learning uncovered truths that escaped the notice of philosophy, psychology and biology?

On one hand, it can be argued that machine learning has, at the least, provided grounds for some of the claims of philosophy regarding the nature of knowledge and its acquisition. Against pure empiricism, induction requires prior knowledge, if only in the form of a constrained hypothesis space. In addition, there is a kind of conservation law at play in induction. The more there is *a priori* knowledge, the easier is learning and the less data is needed, and *vice versa*. The statistical study of machine learning allows quantifying this trade-off. One corollary is that if a cognitive agent is ready to accept any theory of the world, then it becomes unable to learn by induction from data alone. This is deeply connected to Popper's claim that science is characterized by the possibility of refuting its theories. If everything is conceivable, then for any set of data there exists a model for it, and one can no longer avails oneself of the good fit between the model and the learning data to guarantee its relevance for future unknown events.

On the other hand, the algorithms produced in machine learning during the last decades seem quite remote from what can be expected to account for natural cognition. For one thing, there is virtually no notion of knowledge organization in these methods. Learning is supposed to arise on a blank slide, albeit on a constrained one, and its output is not supposed to be used for subsequent learning episodes. Neither is there any hierarchy in the "knowledge" produced. Learning is not conceived as an ongoing activity, but rather as a one shot process more akin to data analysis than to a gradual discovery development, or even to an adaptive process. Indeed, this timeless point of view on learning resides also at the core of the current theories of induction. Thus, the theory that establishes a link between the empirical fit of the candidate hypothesis with respect to the data and its expected value on unseen events becomes essentially inoperative if the data are not supposed to be independent from each other. This requirement is obviously at odds with most natural learning settings where either the learner is actively searching for data or where it learns under the teaching of a tutor carefully choosing the data and their order of presentation. If only for this reason, it would be interesting to remove the blinders of the current dominant theory in machine learning and study learning as a historical process.

To sum up, even if the successes of machine learning so far can, with justice, be considered as impressive, they are nonetheless too often limited to a narrow range of situations. Thus, learning is seen as a passive process with data arriving independently of each other from a stationary environment. Learning is a one-shot activity with training occurring once before testing. Finally, the current theories about learning are mostly worst-case analyses, they suppose that the environment behaves as an adversary trying to conceal its regularities from the learner. In addition, they are unable to account for incremental learning.

From this severe assessment, it seems that machine learning is not yet in a position to offer vistas on incremental learning and on ordering effects. We will however try in the following to extract relevant seeds from recent inquiries in machine learning.

3. Empirical findings : data + sequences = variations

Even though machine learning has not overly concerned itself with the ongoing character of learning, practical considerations sometimes impose resorting to incremental learning. For instance, there are situations where the

data is not entirely available before decisions must be made causing learning to be distributed over time stages. But even in cases where the whole data set is on hand, it can happen that, due to its size, the learning algorithm cannot practically handle it all, and must split it in smaller samples, starting by finding regularities in the first sample, and then use others sequentially to check and modify if needed what regularity has been found. There is, therefore, a collection of empirical findings about what is called *on-line learning*.

One of the most documented and studied spectacular effect is observed with artificial neural networks. These systems are made of numerous elementary computing elements called "neurons" that are often organized in successive layers through which information flows from input units to output units via connexions (see figure 2). In supervised learning, input and output units are repeatedly clipped with the values associated with the training examples, and the strength of the connexions in the network are progressively modified using a gradient descent technique so as to allow the reproduction of the correct output for each example's input. If the neural network has been well-tailored, and if the data are representative of the phenomenon, then generalization can occur from the particular training set to unseen inputs.

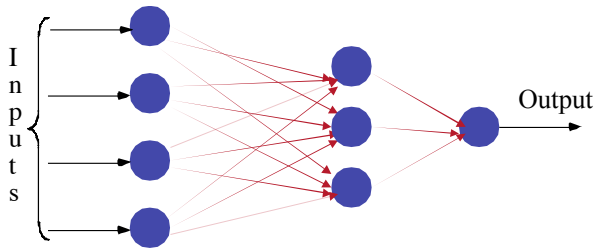

Figure 2. Example of a feed-forward neural network. The input signal is transmitted from one layer of "neurons" to the next one via weighted connexions.

In neural networks, there are two types of learning algorithms. Those that take the whole training set in consideration to compute a total signal error that is used for the gradient descent method, and those that compute local error signals, one for each example in turn. In both cases, the training set must be presented many times to the system before convergence of the connexions strengths occurs. But what happens then if new instances become available? Can we continue learning by concentrating on these new instances without repeating the former ones as, it seems, is routinely done in natural brains? The answer, obtained through numerous experiments, is a resounding no. When an artificial neural network is exposed to new examples, it tends to completely forget what has been learned before, a phenomenon aptly named "catastrophic forgetting" (see French (1997)). Something in these artificial neural networks wreaks havoc. Researchers have attempted to remedy this by acting in two directions: either by manipulating the data and repeating at least part of the former data in the new data sets presentations, or by modifying the learning algorithm itself. It has been postulated indeed that one reason for this undesirable behaviour is the distributed character of the memory in neural network. By somewhat limiting this distributed processing, a more graceful behaviour is obtained in incremental learning.

These efforts however are partial and only scratch the surface of important questions. For instance, which information, in on-line learning, is transmitted from one state of the learner to the next one? More intriguingly, what happens if two pieces of data are presented in two different orders? Is the result of learning the same?

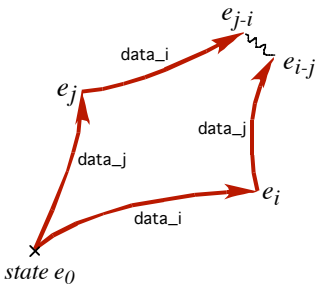

Figure 3. From an initial state e_0, two orderings of data_i and data_j can lead in general, to two different states of the learner. Which information then should be transmitted from one state to the next one so that $e_{i\text{-}j} = e_{j\text{-}i}$?

One can imagine that if the learner purely accumulates its past experience, then no forgetting occurs. The learner passes all this information from one state to the next one, adding in the process more and more information as training data is made available. Then, of course, the system is insensitive to the order of presentation of the data. However, surely, learning can involve much more sophisticated mechanisms than pure memorization of data. Is it still possible to identify the "current of information" flowing from one state to the next? Can we determine the conditions that are necessary or sufficient or both, regarding the data and/or the learner, for order independence in learning?

Incremental learning systems other than neural networks have been devised out of curiosity or for reasons of applicability to real-world problems. For instance, incremental versions of decision trees inductive algorithms (see Utgoff (1989, 1994)) have been published and tested. In unsupervised learning, many algorithms behave incrementally by selecting "seeds" among the training data from where groups or hierarchical categories are built. Always the yardstick against which these systems have been compared is the non incremental version. Variations in the learning output resulting from the order of the training data presentation, also called ordering effects, accompanying these incremental learners have been considered as deviations from the true best obtainable learning results. Thus research efforts have been turned toward the reduction of these effects. It did not occur that the sequence of the data itself can encode useful information that only learners prone to ordering effects could detect, and that non incremental learning is not necessarily an ideal. The same outlook also dominates theoretical studies on on-line learning, the question being there to measure the "regret", or the difference in performance, between an on-line learner and the corresponding off-line one submitted to identical training data.

In this frame of thought, various heuristics have therefore been proposed in order to reduce ordering effects. Foremost among them is the proposal that training examples be presented in as decorrelated an order as possible sequence. For instance, in unsupervised learning, one should try to present examples of each category in turn so that a category does not establish itself too strongly before instances of other categories are observed. But a more radical solution is, rather than to play on the training data order, to modify the incremental learning algorithm. One solution is indeed to keep in memory all past data and recompute at any time the current optimum given all the data observed so far. For instance, the decision tree inducer ID5 (see Utgoff (1994)) is a on-line version of the standard induction algorithm ID3 (see Quinlan (1986)), where enough information about past data is kept in memory to ensure that the same tree as would be induced by ID3 is produced by ID5. The system ID4 which does not memorize as much information as ID5 is, on the other hand, prone to ordering effects (see Utgoff (1989)).

One can see, therefore, that there is an interplay between the training data and the learning algorithm that affects the importance of ordering effects.

4. Sequencing effects and learning

Given a learning task defined by a space of potential examples together with their attached output values (in case of supervised learning) and a cost function over the prediction made by the learner, ***sequencing effects*** arise when the result of learning is affected by at least one of the following parameters: (i) *sampling*, that is the choice of the training data, (ii) the *order* of presentation of these data, and (iii) the *speed* of presentation.

Sequencing effects abound in natural learning situations. It is obvious to any teacher that the learning material, the order in which it is presented, and the speed of presentation of this material, are all of paramount importance in the making of a good lesson. Not all students are equally affected by the possible variations of these parameters, but, at least to some degree, all are. This seems equally true in animal learning. In fact, it is difficult to envision any learning process that would not be sensitive to sequencing effects, that is, plainly, to its history.

It is clear why learning must, in general, be sensitive to sampling, that is, to the choice of the learning inputs. Indeed, following what has been said earlier on inductive supervised learning, each candidate hypothesis is evaluated with respect to the available training data via the inductive criteria. Usually, when one changes the particular training set, one changes also de facto the respective values of the hypotheses, possibly leading to different choice of the optimal hypothesis. In technical terms, this is called the variance. It is usually the case that the variance is directly linked to the richness of the hypotheses space, increasing when the richness increases.

The dependency of learning on the order and the speed of presentation of the data calls for other considerations that can be related to the search procedure used for finding hypotheses that are good under the inductive criteria. Two main parameters can be identified in this respect. The first one is associated to the *memorization of the information contained in the data observed so far*. If no information is discarded, then, in principle, it is possible to restore the entire collection of the past data, and therefore to cancel the effects due to the order of presentation of the data. On the other hand, forgetting aspects of the past data can prevent one from finding the same optimum of the inductive criteria since it is now computed from different data. Since forgetting is usually a function of the order of the data presentation, and possibly of its duration, the result of learning can be affected as well by the same factors.

The second parameter is linked to the *properties of the search procedure* used for the exploration of the hypotheses space. Even in the case that no information on the training data is lost, if the search procedure is not guaranteed to return the optimal hypothesis, but, for instance only a local one, then the order in which the data is processed can lead to different end results. This is illustrated in figure 4 for a gradient method. If, in addition, the search procedure exhibits inertia due to limits on computational resources, then the order and speed issues acquire even greater importance.

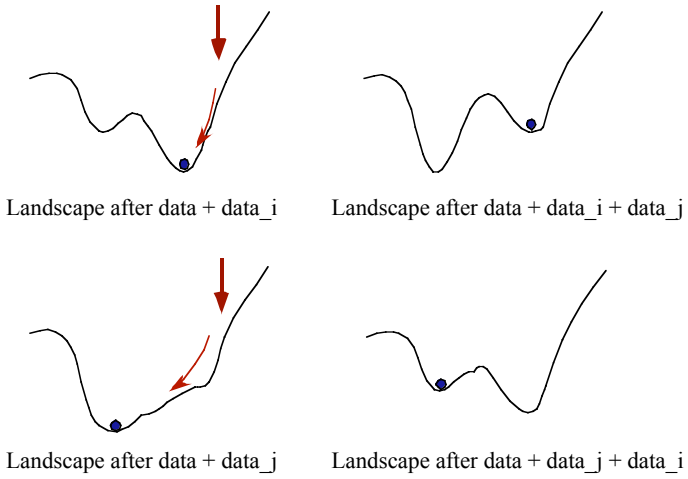

Figure 4. The optimisation landscape is changed with each new piece of data, and may differ depending on the order of presentation of the data.

Sampling, order, and speed of presentation thus define a three dimensional control parameter space (see figure 5). A number of research forays in machine learning explore this space, even if, admittedly, in a still very limited way. This is the topic of the next section. Along the way, it will be interesting to consider a set of questions and examine which light, if any, machine learning research brings on them.

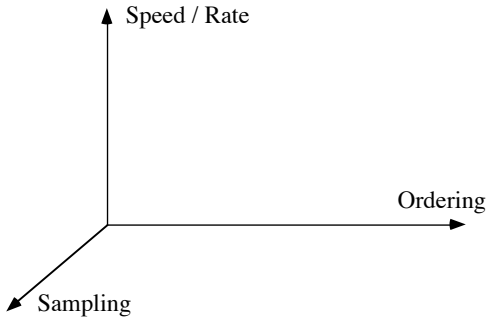

Figure 5. The three dimensional space of the control parameters for ordering effects in incremental learning.

These questions include obvious questions on the controllability of learning:

- Are there general strategies for determining a priori, or on-line, the most appropriate data sample, order and speed of examples presentation?
- Are there advantages to ordering effects? And, if yes, how to profit from them?
- More specifically, does the choice of data sample, order and speed of presentation allow learning more? or simply learning more easily?
- Under what conditions, either on the data or on the learner, can incremental learning be rendered order-independent?

There are also questions related to the learning process as an ongoing activity:

- Are there harmful pieces of data? What characterizes them? How do they affect learning? How do learners recover from them?

- What are the conditions for positive or negative transfer from one learning step or task to the next one?
- Is it sometimes necessary or useful to retrace steps in learning, for instance, reconsidering past hypotheses or past data sets?
- Would it be possible to infer an unknown concept in such a way that only better and better hypotheses are inferred? And, more generally, can monotonic learning be always successful?
- How does prior knowledge affect sequencing effects?
- What kind of relationships, if any, exist between the information content of the data, the training sequence, the properties of the learner and ordering effects?

5. Controlling sequencing effects: a machine learning perspective

Before considering ways of controlling incremental learning, a preliminary question needs to be addressed: how to measure or evaluate different learning trajectories?

Under the classical model in machine learning, only the end result of learning matters, the vagaries during the learning process are deemed of no consequences. But this point of view seems too limited in the study of incremental learning since one aspect of interest is the easiness of learning, not merely the end product or even the rate of convergence to the final hypothesis or to a final level of performance. Several other performance criteria have thus been defined in order to assess the merits of various learning strategies.

Some merely quantify the degradation of performance compared to one-shot, non-incremental learning, resulting from the lack of information during learning. This is the "regret" notion. It is usually cast into a game theoretical framework where one compares the gain of a set of various "players", i.e. learners, over a given sequence of training data.

Other criteria quantify the whole learning trajectory underlining specific aspects. For instance:

- The number of needed training instances.
- The number of mistakes of the learner made during training. This mostly applies in concept learning tasks where the learner tries to identify a target concept. For each new input, the learner is first asked to give its class according to its current knowledge, before being given the correct answer. Each mistake is counted.
- The number of mind changes of the learner during training. In the machine learning perspective, a *mind change* occurs whenever the learner changes its mind about the best hypothesis or model of the data observed so far.
- The memory size required during learning.

It is then possible to analyze learning settings where either the data sample, or the order or the speed of presentation, or a set of parameters controlling the learners, are varied.

Active learning

In active learning, the learner is no more passively waiting for the data to come, but it has, to varying degrees, some autonomy in the search for information.

In the weaker setting, the learner can only choose examples among a predefined training data set. For instance, this could apply to the case where data are abundant, but the cost of labelling each example is high so that only the data selected by the learner will be submitted to the expert. Two cases may arise, depending on whether the learner must make its choice prior to learning, or if it can make its decision piecewise during learning.

In the first case, this amounts to changing the input distribution. Since the selection of the more useful data must be done before learning, the choice is usually done on the basis of a crude preliminary data analysis phase. For instance, one may try to eliminate redundant examples, or to discard seemingly irrelevant attributes describing the examples. Or one may want to detect and retain prototypes, for instance, centres of groups of data points, in the hope that these encode most of the information without be overly affected by noise or spurious effects in the original data sets. The question is whether such a "filter" method, roughly independent of the learning algorithm, can be devised in a general way. It is in fact not the case, and no filter is universally useful. It all depends on the kind of regularities that are searched in the data.

In the second case, the selection of the next data point(s) to query is made on the basis of the current candidate hypothesis. In concept learning, there are two main approaches. One is to modify the distribution of the input data from one run to the next, as in the boosting method. In a way, this is not really incremental learning, but rather an iterative process where more and more weight is given to examples that were not correctly classified in the former run. The final decision function results from a weighted sum of the hypotheses learned at every step. The second approach properly belongs to the incremental scheme with new examples processed on-line. Again, one can distinguish two principal mechanisms. The first one is to try to gain as much information as possible with each new data point. This can be done simply by selecting data points in regions where one does not have yet data, or where the performance so far is bad, or where previously found data resulted in learning. One can also try to measure the uncertainty affecting the current candidate hypotheses and select data that reduce it as much as possible. Technically, this can be done in several ways. For instance:

- One can try to reduce the number of remaining candidate hypotheses, what is called the version space in concept learning. The difficulty here is first to evaluate the "volume" of the space, and second to identify one optimal example, that is, one that allows eliminating approximately one half of the version space.
- Another solution consists in measuring the disagreement among hypotheses sampled from the current version space on potential training examples, and to retain one example for which this disagreement is maximal.
- A third method, not limited to concept learning (i.e. only two classes), is to compute the entropy of each potential class for each remaining instance and each possible labelling of them. One then selects the instance that maximally reduces the entropy. This can be computationally expensive.

There are other proposed schemes, but all revolve around the idea of trying, one way or another, to reduce as much as possible the uncertainty about the remaining candidate hypotheses.

Another approach does not take into account the whole version space, but focuses instead on the current candidate hypothesis. In this framework, one tries to get more details about the hypothesis where differences in prediction can be more critical. For instance, when the hypothesis can be seen as a decision function separating the space of examples in two regions, there can be regions where positive and negative instances are closed to each other and where, therefore, it is crucial that the decision boundary be well-informed. Active learning consists then in selecting data points close to that estimated boundary in order to accurately define it.

One borderline case is the one of reinforcement learning. There, an agent perceiving (part of) its environment and choosing actions that modify its state in the environment, tries to identify courses of actions, called policies, that maximize a cumulative gain over time. Initially the agent knows very little about the environment and it must learn its characteristics while acting. In order to discover an optimal or near-optimal policy, the agent must usually control a trade-off between exploring the world around, so that it does not misses valuable opportunities, and exploiting its current knowledge about it. Here the data points correspond to states of the agent in the world, and they must be tested in order to discover what opportunities they offer in terms of gain. It was found that, under some stationary conditions about the environment, the optimal sampling strategy is to greedily select the seemingly most promising states, except for a small diminishing proportion of time. Strictly, this guarantees correct learning only in the limit of infinitely many visits of each state. More interesting are the studies of learning games using reinforcement learning. There the question arises as to what is the best progression of "teachers". Should one start by confronting novice players before playing against experts? Or should there be a period of self-play: the machine against the machine? No definite answer has been found, even if it is generally observed that facing players of increasing competence is better than the other way around. This question more properly belongs to the general issue of teachability discussed below.

Finally, other active learning paradigms have been investigated, specially in theoretical terms. In one, called the "membership query" model, the learner queries any data point and is answered whether the queried example belongs or not to the concept (see Angluin (1988)). In the "equivalence query" model, the learner can ask if one hypothesis is the correct one, and the "oracle" either answers "yes" or provides a counter-example (see Angluin (1988)). One interesting result is that if the learner can test hypotheses that are outside the set of target concepts, then learning can possibly require less training data than if it is required to stay within this set.

Overall, when active learning consists in selecting examples to label from a given set of unlabeled data, theoretical results show that the number of training data can be reduced substantially if they are selected carefully. In some cases, this can make the difference between tractable and intractable learning. However, these studies only look at the acceleration of learning, not at differences in what can be learned with different orders of presentation of the training data.

From a conceptual point of view, the study of active learning is very important because it forces researchers to turn away from the assumption that data are identically and independently distributed, which is at the basis of most of the theoretical constructions so far. One can expect radically novel ideas when this turn will be effected.

The control parameters of the learner

When this is not the learner itself that controls a part of the learning process, but an external agent, there exists two broad classes of possible interferences with the learning process. Either one tries to change the learner's characteristics, or one tries to control the sequence of inputs. The latter case, called teachability, will be dealt with latter.

There are many control parameters to a learning system. The question is to identify, at a sufficient high-level, the ones that can play a key role in sequencing effects. Since learning can be seen as the search for an optimal hypothesis in a given space under an inductive criteria defined over the training set, there appears readily three means to control learning. The first one corresponds to a change of the hypothesis space. The second consists in modifying the optimisation landscape. This can be done either by changing the training set, for instance by a forgetting mechanism, or by changing the inductive criteria. Finally, one can also fiddle with the exploration

process. For instance, in the case of a gradient search, slowing down the search process can prevent the system from having time to find the local optimum, which, in turn, can introduce sequencing effects.

Not all of these possibilities have been investigated in machine learning. Mostly, there have been some theoretical studies dealing with the impact on learning of such quantities as:

- the memory capacity, and particularly the trade-off between short-term memory and long-term memory
- the number of mind changes allowed to the learner before it has to settle for a definitive hypothesis
- the number of errors allowed during learning
- the computational resources

Some studies have also dealt with the effect of imposing some monotonic constraint on learning, such as forbidding to allow considering new candidate hypotheses that are in error on some past data points.

Overall, the findings are generally valid only in very restricted conditions, and it is difficult to interpret or generalize them. One can cite for instance the result that, in inductive concept learning, order independence is only possible if no forgetting of the training data occurs. This underlines the fact that order independence is in fact a very stringent and unreasonable restriction on learning.

How to optimize the learner given that the ordering and the target concept are known in advance? Or, what is the optimal ordering for a given learner? Those are questions that do not have answers yet.

Teachability

In teachability studies, the problem is how to select the best training data and/or the order of presentation such that learning of a given target concept or target class of concepts is facilitated. One question in particular is to quantify the minimal sequence of training data needed to teach a target concept to a learner (see Goldman & Mathias (1996)). This can be in terms of the sample size required, or of the minimal number of mistakes the learner is to make before reaching the correct hypothesis. Obviously, the notion of a learner has to be more precisely defined. It can be some given learner, the characteristics of which are known to the teacher, or it can be any learner, or the worst possible one. In these cases, the learner should still obey some constraints. In machine learning, one often imposes that the learner only considers hypotheses that are coherent with the data, for instance, that they cover all positive instances and no negative ones.

The size of the minimal training set needed to teach a coherent learner is called the “teaching dimension”. It obviously depends on the types of laws that Nature and the learner may consider. For instance, if the possible hypotheses take the form of convex rectangles in 2-dimensions, and if the learner always outputs the most general hypothesis consistent with all past training inputs, then it suffices to provide the learner with the two inputs associated with two opposites vertices of the target rectangle. It is therefore apparent that the problem is to determine the right encoding of the target law within the instance language and given some knowledge about the learner's functioning.

While relatively few empirical works have been reported on this question, theoreticians have pondered with anguish on the line separating teaching from cheating. One can indeed see teaching as a kind of encoding operation by which a teacher tries to transmit some information to the learner using a coded message: the training data. If however nothing prevents the teacher and the learner from having agreed beforehand on some private code, then it is easy for the teacher to transmit the identity of the hypothesis in this secret code, thus

completely bypassing a true learning mechanism. Considerable thinking has therefore been spent on cleverly defining the rules of the teaching game so as to prevent any possibility of collusion. One must acknowledge that, overall, this has been detrimental to the search for reasonable scenarios and to obtaining interesting results. In fear of forbidden collusion between learner and teacher, the theoretical settings devised by researchers have actually prohibited any possibility of a constructive cooperation and only address scenarios where the learner becomes obstinate and adversarial, trying as hard as possible not to learn! In the present state of affairs, waiting for renewed theoretical approaches, the more interesting ideas come from the study of heuristics designed for empirical experiments.

One early proposal is to select for teaching what are called “near-miss” examples (See Winston (1970)). These examples (positive or negative) differ from the current best candidate hypothesis by as few relevant aspects as possible, the idea being to help the learner to focus on relevant differences between the current hypothesis and the target concept. The difficulty here, in order to transfer that principle to natural learning, is that a perfect knowledge of the learner and of its current set of candidate hypotheses is required. Other proposals are described in more general terms. For instance, one idea is to provide the data in increasing order of complexity. One could translate this into “teach simple things first”. However, the question remains to define a reliable complexity criterion. Some concepts can be easy to one learner, while being unfathomable to another or to the same learner at another time. Most current attempts define complexity in terms of superficial syntactical complexity. This certainly is unsatisfactory, and there remains a long way to go between such crude criteria and the principles used for instance in educational science where concerns for such things as the historical genesis of the subject matter or its epistemology are everyday life. There is nonetheless one promising research direction called hierarchical learning (see Barto and Mahadevan (2003), Murphy and Lassaline (1997), Rivest and Sloane (1994)). The idea is to teach sub-concepts before higher knowledge structures. This has been tested for the learning of concepts expressed in first order logic where one can identify logical sub-expressions. Once again, the teacher must know the definition of the target concept, and at the same time identify sub-concepts and relevant corresponding training data.

As one can see, teaching requires knowledge of the domain, of the learning target, and of some characteristics of the learner. This suggests a trade-off between the complexity of teaching and the complexity of learning. This is one open question among many for which there are not, as yet, answers from the study of machine learning. We are far from being able to shed some light on the teaching of Newtonian mechanics or the French grammar to human students or artificial learners.

6. Conclusions

There are many reasons why artificial intelligence need to study sequencing effects. First, natural learning systems learn over time, be them human or not, and all seem prone to sequencing effects, in particular related to the order and speed of the learning inputs and learning tasks. If artificial intelligence is to be part of the overall scientific effort towards understanding cognition and in particular natural cognition, it cannot ignore these aspects of learning. Second, if only on practical and engineering grounds, artificial intelligence must increasingly deal with incremental learning. The huge databases now readily available can no longer be handled in one shot. They have to be analyzed in a piecemeal fashion, and hence, at least partially, sequentially. This implies that choices will have to be made regarding the selection of training data subsets and their order of presentation. Likewise, many learning systems are now embedded in long-life computer systems and must confront sequences of inputs, drifting environmental conditions, and evolutive tasks. Per force therefore,

artificial intelligence engineers will become increasingly aware of sequencing effects, and will have to find intelligent ways to cope with them. Finally, there are reasons to believe that deep issues in cognition and information theory are connected with the study of incremental learning and sequencing effects. For instance, there has been much debate about the redundancy in a set of information. But this question of redundancy becomes even more interesting when it is discussed in the context of a sequence of information, because then the place in the sequence has to be taken into account, and even more, the possibility for misleading pieces of information must be examined.

While it has been always considered that a piece of information could at worst be useless, it should now be acknowledged that it can have a *negative* impact. There is simply no theory of information at the moment offering a framework ready to account for this. Related to this issue is the problem of transfers, positive or negative, between learning tasks. This is still very little studied in artificial intelligence, and only in the context of independent computer simulations for very limited sets of learning tasks involving relatively poor prior knowledge. It is obvious that if one is seriously interested in understanding and controlling transfer in learning, for instance in the hope to enhance the efficiency of the educational system, much more research is to be done in this area.

More generally, the study of incremental learning and sequencing effects should bring new light on fundamental questions such as : What is the impact of forgetting ? Can it be helpful in some cases ? What kind of data should preferably be dispensed with in case of memory size constraints ? What is the best teaching strategy ? What is the tradeoff between the complexity of learning and the complexity of teaching ? And so on.

With regards to these questions, there are two ways to assess the current status of machine learning research. The first one is pessimistic and gloomily considers the blindness of the current dominant paradigm in machine learning concerning incremental learning and sequencing effects, as well as the lack of interest so far from the practitioners. In addition, machine learning so far is mostly limited to the learning of one concept, using a single learning mechanism with little prior knowledge. This is indeed quite a poor framework for the study of something as complex as learning can be in natural systems. The second way is optimistic and sees this state of affairs as an opportunity to open up new and interesting directions for research, that could have a major impact on the development of cognitive science as a whole. If it is true that machine learning research has little to offer as yet regarding on-line and long-life learning, it remains that it offers tremendous opportunities for research in this area, if only because artificial learners and artificial contexts are more easily controllable than natural ones.

In effect, machine learning research has already brought us several interesting concepts. Most prominently, it has stressed the benefit of distinguishing between the properties of the hypothesis space —its richness and the valuation scheme associated with it— and the characteristics of the actual search procedure in this space, guided by the training data. This in turn attracts attention towards two important factors related to sequencing effects, namely forgetting and the (non)optimality of the search procedure. Both are key parameters than need to be thoroughly understood if one is to master sequencing effects. Indeed, in retrospect, it is hard to believe that learning and memorization can be studied without regard to forgetting. There is thus no doubt that sequencing effects are deeply associated with the fundamental properties of learning. They therefore deserve to be actively examined.

It is foreseeable that machine learning research will increasingly turn toward the study of incremental learning. Among the reasons for this new focus are the current interest for text mining, and what are texts if not complex

sequences of data, the growing concern for computer aided education, and the design of more complex learning systems embedded in long-life artificial systems.

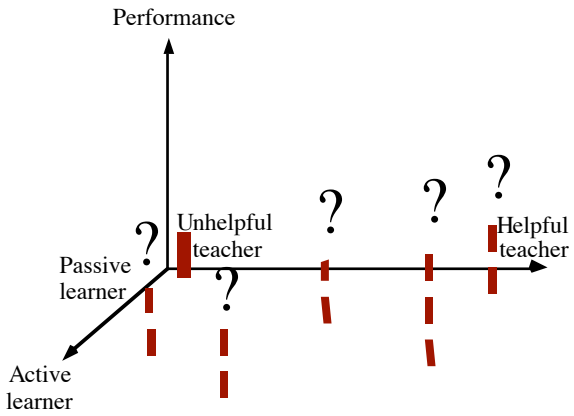

Figure 6. So far, in machine learning, we have performance measures only for passive learners and unhelpful teachers. Future work should provide us with more insights on other mixtures of learning parameters.

There is hope, therefore, that someday in a not too far future, we will be more knowledgeable about sequencing effects, learning trajectories and the design of teaching sequences. We should be careful, however, and not be overly optimistic. It will never be easy to craft optimal or even good teaching sequences. After all, it is quite difficult to find a sequence of appropriate operations for transforming a Rubik's cube back to its original configuration. Finding a sequence of inputs that carefully guides a complex learner toward some goal state will be at least as difficult. In some sense, education and the Kibur's megagame are not that different from the Rubik's cube game. We have thus a lot of fun and frustration waiting for us.

Projects

1. Explain why the candidate elimination algorithm of Mitchell (1983) is not subject to the order of presentation of the data

Propose changes in the algorithm that would make it order sensitive?

2. Use the SNNS software freely available on the Internet () to learn the letters in the data set provided in the letters.pat file available with the software. Learn first the whole set of the 26 letters. This should require typically a few hundreds epochs to get an accuracy better than 1%.

In a second stage, learn first one half of the 26 letters, and only when it is learn with high accuracy, learn the second half of the letters. What do you observe on the recognition of the letters learnt in the first stage?

Test other incremental learning protocols (e.g. learn the first half, then the second, then the first half again; or learn the letters in sequence, presenting each one of them at least 100 times before processing the next one). What do you observe? Analyze what could explain these behaviours. What could be a good strategy to learn the whole set when the first half has to be learnt in a first stage ?

3. Human learners are subject to ordering effects also. For instance, if asked to split a square in 5 equal parts, human subjects answer quickly and without hesitation (see figure 7 (left)).

On the other hand, they have considerable trouble if they have been asked beforehand to solve the following sequence of problems : split the amputated square (see figure 7 (right)), first in 2 equal parts, then in 3 equal parts and then in 4 equal parts.

Replicate this experiment with your friends.

Try to understand the reasons that can explain this behaviour. Try then to invent other sequences of tasks that can produce significant order effects.

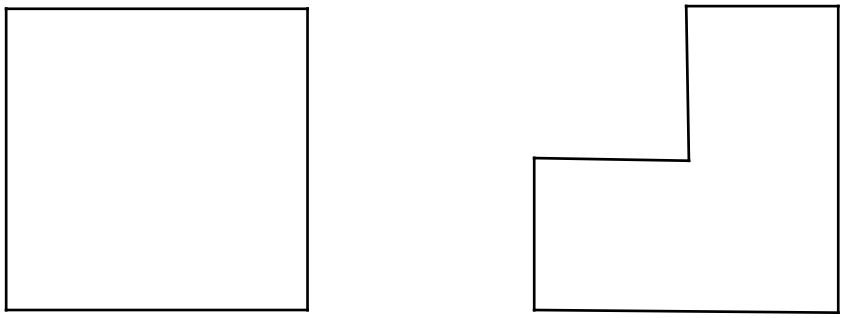

Figure 7. A square that must be split in 5 equal parts (left). The amputated square (one fourth has been removed) that has to be split in 2, 3 and then 4 equal parts (right).

References

Angluin, D. (1988): Queries and concept learning. *Machine Learning Journal*, **2,** 319-342.

Barto, A. and S. Mahadevan (2003): Recent advances in hierarchical reinforcement learning. *DiscreteEvent Systems journal*, 41-77.

Cornuéjols, A. & Miclet, L. (2002) *L'apprentissage artificiel. Méthodes et concepts*, Eyrolles.

French, R. (1997): Catastrophic Forgetting in Connectionist Networks. *Trends in Cognitive Sciences*, **3,** 128-135.

Freund, Y. and R. Shapire: A tutorial on boosting. [Available online from www.research.att.com/~yoav.www.research.att.com/~schapire.]

Goldman, S. and D. Mathias (1996): Teaching a smarter learner. *Journal of Computer and System Sciences*, **52,** 255-267.

Minsky, M. and S. Papert (1988): *Perceptrons (2nd ed.)*, MIT Press.

Mitchell, T. (1997) *Machine Learning*, McGraw-Hill.

Murphy, G. and M. Lassaline (1997): Hierarchical structure in concepts and the basic level of categorization. *Knowledge, concepts, and categories*, K. Lambert and D. Schanks, Eds., Hove: Psychology Press.

Opper, M. (1999). A bayesian approach to online learning. In D. Saad (Ed.), *On-line learning in neural networks* (pp. 363-378), Cambridge University Press.

Quinlan, J. (1986): Induction of Decision Trees. *Machine Learning Journal*, **1,** 81-106.

Rivest, R. and R. Sloan (1994): A formal model of hierarchical concept learning. *Information and Computation***,** 88-114.

Rosenblatt, F., (1962): *Principles of neurodynamics: Perceptrons and the theory of brain mechanisms*. Spartan; Washington, DC.

Russell, S. and P. Norvig (2002): *Artificial Intelligence. A modern approach (2nd ed.)*, Prentice Hall.

Schölkopf, B., Burges, C. & Smola, A. (1999) *Advances in Kernel Methods. Support Vector Learning,* MIT Press.

Utgoff, P. (1989): Incremental Induction of Decision Trees. *Machine Learning Journal*, **4,** 161-186.

Utgoff, P. (1994): An Improved Algorithm for Incremental Induction of Decision Trees. *Int. Conf. on Machine Learning (ICML-94)*, Morgan Kaufmann, 318-325.

Vapnik, V. (1995) *The nature of statistical learning theory,* Springer-Verlag.

Winston, P., (1970): Learning structural descriptions from examples, [MIT Technical Report AI-TR-231], MIT.