A Note on the Use of Probabilities by Mechanical Learners*

Eric Martin¹ and Daniel Osherson²

¹ IDIAP, C.P. 592, CH-1920 Martigny, Switzerland (martin@maya.idiap.ch)
² Dipasco, Istituto San Raffaele, Via Olgettina 60, I-20132 Milano, Italy (osherson@idiap.ch)

Abstract. We raise the following problem: in a probabilistic context, is it always fruitful for a machine to compute probabilities? The question is made precise in a paradigm of the limit-identification kind, where a learner must discover almost surely whether an infinite sequence of heads and tails belongs to an effective subset S of the Cantor space. In this context, a successful strategy for an ineffectual learner is to compute, at each stage, the conditional probability that he is faced with an element of S, given the data received so far. We show that an effective learner should not proceed this way in all circumstances. Indeed, even if he gets an optimal description of a set S, and even if some machine can always compute the conditional probability for S given any data, an effective learner optimizes his inductive competence only if he does not compute the relevant probabilities. We conclude that the advice “compute probabilities whenever you can” should sometimes be received with caution in the context of machine learning.

1 Overview

1.1 Goal of this study

The probability calculus appears to be an ideal instrument for verifying or disconfirning hypotheses in settings involving random processes. In such settings it is tempting to design mechanical learners that compute the probabilities of rival hypotheses, assuming that the needed probabilities can in fact be computed. The purpose of the present note is to show that — the probabilistic temptation notwithstanding — there are classes of learning problems X in which (a) probability calculations suffice to solve every member of X, (b) the needed probabilities can be computed by machine, but (c) it is not advisable to design a learning algorithm for X based on probability. The result may help to sharpen reflection about the use of probability in mechanical reasoning, since by (b) there is

* We thank Scott Weinstein, Michiel van Lambalgen, and several anonymous reviewers for helpful comments. Research support was provided by the Office of Naval Research under contracts Nos. N00014-87-K-0401 N00014-89-J-1725, by the Swiss National Science Foundation under grant number 21-32399.91, and by the Centre National de la Recherche Scientifique.
no computational obstacle to reasoning probabilistically. Regarding (c), it will be shown that although $X$ is probabilistically solvable by machine, there is a second class of problems $X'$ with the following properties. Just like $X$, $X'$ can be probabilistically solved by machine. Moreover, some mechanical learner can solve all the problems in $X \cup X'$. However, no algorithm that proceeds probabilistically on $X$ solves all of $X'$, probabilistically or otherwise. Hence, proceeding probabilistically on $X$ limits scientific competence, notably, with respect to $X'$.

In the present section we offer an informal explanation of our result. Formal details are presented starting with Section 2.

1.2 Inferring properties of coin-toss sequences

To define $X$ and $X'$, we rely on the setting of fair coin-tosses. The learner's task is to indicate whether an infinite sequence of tosses belongs to a prespecified, measurable subset $S$ of the space of all such infinite sequences. Following each observed toss, the learner issues a number in the unit interval. Success is defined as convergence to 1 if the infinite sequence lies in $S$, and convergence to 0 otherwise.

**Example 1.** Consider the set $S^*$ holding every infinite sequence that begins with 100 heads. Then, faced with a finite sequence $\sigma$, the learner might choose to issue:

\[
\begin{align*}
0 & \quad \text{if a tail occurs among the first 100 positions of } \sigma, \\
2^{\text{length}(\sigma)} - 100 & \quad \text{if } \text{length}(\sigma) \leq 100 \text{ and } \sigma \text{ contains no tail}, \\
1 & \quad \text{otherwise.}
\end{align*}
\]

Such problems are easily solved with probability 1 if viewed ineffectively. In this case, it suffices to issue the conditional probability of $S$ given the finite sequence of toss-outcomes observed so far. In contrast, formulating an effective version of the paradigm is more delicate since the inputs and outputs to the learner must be finitized in some appropriate way.

Regarding outputs, we allow mechanical learners to issue indexes for recursive reals, that is, indexes for Turing Machines that enumerate the decimal expansion of a number in $[0, 1]$. For example, the probabilities arising in Example 1 are recursive. Regarding inputs, two things must be communicated to the learner, namely, the subset $S$ defining the problem, along with the current data. The latter poses no problem, since the data always form a finite sequence of heads/tails. We use $\text{SEQ}$ to denote the collection of all such finite sequences. The remaining question is how to communicate the set $S$. This will be achieved via indexes for sets in the effective Borel Hierarchy, whose basic concepts are now reviewed. [More complete treatment is available in Hinman (1978).]

1.3 The effective Borel Hierarchy

Let $S$ be a collection of infinite sequences of coin-tosses. $S$ is classified as $\Sigma_0^0$ and also as $\Pi_0^0$ if there is a Turing Machine $M$ with the following property: When $M$ is started with the infinite sequence $b$ on its input tape, $M$ eventually halts