SIMULATION AND FORMAL MANIPULATIONS: COMPLEMENTARY APPROACHES FOR STUDYING RANDOM EVENTS WITH COMPUTERS

Jacques Bordier, Télé-Université, Canada
Anne Bergeron, Université du Québec à Montréal, Canada

In this paper we discuss computer environments that perform both simulation and symbolic manipulations. Recent advances in symbolic computations have allowed us to develop a computer laboratory that can handle formally a vast range of random experiments, including experiments with infinite possible outcomes. In the laboratory students can describe experiments using a simple formal language made up of primitives. They then obtain representations of possible outcomes, results of simulations, and exact answers for the probabilities of random events, as well as the values of different statistics.

Even for students with minimal mathematical skills, working with such environments brings back the possibility to confront experimental and theoretical values, which is at the core of a deeper understanding of probability and statistics, and their relation to reality.

INTRODUCTION

Experimentation with random phenomena has been proposed, for a long time, as a method for giving students the opportunity to grasp the relation between results of probability calculations and the reality of such phenomena. However, the time required to get results that have statistical significance is generally very long, and experimentation in a classroom are doomed to end rapidly after a few attempts.

More recently, simulations on computers have been proposed to replace actual experimentations [Bergeron and Bordier 1991, Bordier et al 1994]. Computer simulations are useful for the setting up of many didactic situations but, by themselves, they cannot explain the underlying phenomena. In order to get a deeper understanding, one has to construct the space of possible results of an experiment and analyze it with mathematical tools. On the other hand, except for very elementary experiments, the mathematical tools used in analyzing most situations - even simple ones like tossing a coin repeatedly - are beyond the reach of most students. This is the case for problems asking for the expected waiting time for an event, which are modeled using infinite series. In this paper, we show that for a large class of problems of this sort, it is possible to automate the construction, and the summation of the corresponding infinite series.

COMPUTATIONAL PROBABILITIES

A discrete random experiment can often be described by the set ? of its possible outcomes, and by a function:

$$p : ? \rightarrow [0, 1]$$

assigning a probability $p(e)$ to each elementary event e ??? such that $\sum_{e \in \Omega} p(e) = 1$.

Our goal is to construct a virtual laboratory where one could define, simulate and compute exact answers to problems such as:

How many balls does one have to draw from an urn containing three balls, a red, a blue and a white, before one gets a ball of each color, if drawings are with replacement?

What is the probability that the pattern head-head-tail shows up before the pattern head-tail-tail while repeatedly tossing a coin?

In each of these examples, there is a simple experiment – drawing a ball, tossing a coin – with few outcomes. This elementary *trial* is repeated until the sequence of outcomes have a specific property. These experiments belong to a large class of experiments for which computational tools can be developed.

Representing infinite sets of sequences

The first problem in dealing with infinite sets is to be able to describe them with a simple formalism. Consider, for example, the experiment E_1 of repeatedly tossing a coin until the pattern *head-head-tail* – abbreviated as *hht* – appears. The set of possible outcomes looks like:

$$_1 = \{hht, hhht, thht, hhhht, thhht, hthht, tthht, hhhhht, ... \}$$

In this set, the sequence *hhthht* does not appear since the experiment would have been stopped after the third throw.

The main tool for describing such sets come from automata theory, whose goal is to describe sets of sequences. Basically, a finite automaton is a graph whose oriented edges are labeled with letters corresponding to elementary events (Figure 1).

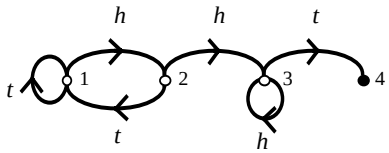

Figure 1: An automaton describing a set $?_1$

Examining Figure 1, we can see that any sequence that goes from node 1 to node 4 ends with the pattern *hht*, and does not contain any other occurrence of this pattern. We say that this automaton *describes*, or *recognizes*, the set 1.

Rational Experiments

We first define a *trial* as a random experiment whose possible outcomes are in a finite set *T*, each result having a fixed probability *p* of occurrence. An *experiment* *E* consists of repeating a trial until a certain condition is satisfied. We say that the experiment is *rational* if the set of its possible results can be recognized by a finite automaton.

For example, the experiment of throwing a coin until both head and tail have appeared at least once is a rational experiment. On the other hand the experiment of throwing a coin until the number of heads equals the number of tails is not rational. It can be shown that no finite automaton can describe all the possible outcomes of this experiment. A more general discussion on the characterization of these experiments will be given in the next section.

Computing probabilities with formal series

In order to be able to answer questions about an experiment, we will associate to the experiment a *formal series* which will contain all relevant information about the experiment. Consider, for example, the experiment E_1, and suppose that the probability of *head* and *tail* is 1/2. Then the probability of a sequence, say *thhht*, is given by $(1/2)^5$, or 1/32. We now consider the formal sum:

$$E_1(h,\ t) = \frac{1}{8}hht + \frac{1}{16}hhht + \frac{1}{16}thht + \frac{1}{32}hhhht + \ldots$$

In this sum, the coefficient of each term is the probability of occurrence of the corresponding sequence. If the expression $E_1(h, t)$ is known, many questions about the experiment can be answered. For example, consider the problem of evaluating the expected number of throws before the pattern *hht* occurs. This number is given by the sum:

$$n = \frac{1}{8}3 + \frac{1}{16}4 + \frac{1}{16}4 + \frac{1}{32}5 + \ldots$$

where each sequence in $E_1(h, t)$ has been replaced by its length. But it is possible to obtain the number n in another way. We first substitute h and t by s in $E_1(h, t)$, yielding:

$$F(s) = E_1(s, s) = \frac{1}{8}s^3 + \frac{1}{16}s^4 + \frac{1}{16}s^4 + \frac{1}{32}s^5 + \ldots$$

We derive $F(s)$ with respect to s, and we evaluate the result for $s = 1$.

All these manipulations will make sense if it is possible to obtain a closed form for $E_1(h, t)$. For rational experiments, the good news is that these functions are always the quotient of two polynomials that is, rational functions. This is done by considering the matrix M whose coefficients $m_{i,j}$ are the labels of arrows from node i to node j in the automaton of Figure 1, multiplied by their elementary probability. Results from graph theory tell us that the matrix M^k describes the sequence of labels of paths of length k between node i and node j. We are interested in all possible paths between node 1 and node 4, that is, the coefficient (1, 4) in the matrix:

$$I + M + M^2 + M^3 + M^4 + \ldots$$

But this matrix is also the inverse of $(I - M)$ where I is the identity matrix. In order to compute $E_1(h, t)$, we need only to inverse the matrix $(I - M)$, and get the value of its (1, 4) coefficient. This kind of inversion is easily carried out by symbolic mathematics software – Maple in our case. We obtain:

$$E_1(h, t) = \frac{1}{4 - (2t + ht)}hh\,\frac{1}{2 - h}t$$

With this form, we easily get the value

$$F(s) = \frac{s^3}{(4 - 2s - s^2)(2 - s)}$$

which can be derived – again with the help of Maple – yielding the value 8 for the expected length of the experiment.

THE LABORATORY

We have given a brief outline showing how it is possible to automatically compute exact answers to probability problems related to rational experiments. Based on the results of (Bergeron, 1992), we constructed a laboratory which we describe in this section.

Defining and Simulating Experiments

In order to define an experiment, the user must specify an elementary trial and the probabilities associated with each outcome, for example drawing a ball from an urn with results {red, blue, white}, each with a probability 1/3. Each experiment in the laboratory will consist in repeating the trial until a certain condition is satisfied.

Describing stopping conditions is done by a language that allows the user to specify either: 1. A pattern, or sequence of patterns, that will stop the experiment. 2. Unions of sequences of patterns. 3. Properties of the sequence of outcomes such as the frequency of a given outcome, the number of different outcomes, or the length of the experiment.

For example, stopping conditions for the urn containing three balls could be: *all three colors have appeared*, or the more complex *a sequence of two red balls eventually followed by a blue ball, or a blue ball eventually followed by a sequence of two red balls.*

It can be shown (Bergeron, 1992) that any experiment generated by using this language gives rise to a rational experiment. With the automaton associated to an experiment, it is easy to perform simulations: using the automaton, the computer generates random numbers and chooses an edge accordingly. Figure 2 shows the cumulative mean length of the experiment of drawing a ball in an urn containing 3 balls until all three colors have been drawn. The graph clearly exhibits a stabilization process.

Once an experiment is defined, the user can define random variables and compute probabilities on them. For example, the *length* of an experiment is defined as the number of elementary trials necessary to complete the experiment. The expectation of this variable would give a theoretical value for the mean length of the experiment.

For the example of the preceding section, the expected length is 5.5, which is corroborated by the simulations of Figure 2.

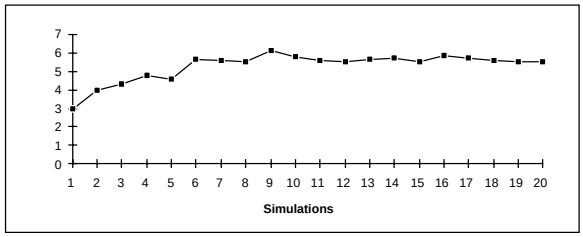

Figure 2. The Variation of Mean Length in 20 Simulations

Interfacing the Laboratory

Even if the laboratory can be used as a stand alone application, we intent to use it as an *expert* underlying pedagogical applications available on the Internet. Smaller

environments can focus on specific problems and provide interfaces suitable to a target group of students. For example, we developed an application that explores various phenomena arising in the experiment of throwing a coin and stopping when one of two patterns occurs. This environment, and others to come, will be made available on the Web as a part of a library of experimental probability applications.

REFERENCES

Bergeron, A., and Bordier, J., (1991). An Intelligent Discovery Environment for Probability and Statistics, in *Advanced Research on Computers in Education*, R. Lewis and S. Otsuki, Eds, North Holland.

Bergeron, A. (1992). Symbolic Computation and Discrete Probabilities, in *Atelier de Combinatoire franco-québecois*, edited by J. Labelle and J.-G. Penaud, Cahiers du LACIM #10, UQAM, 27-42.

Bordier, J., Bergeron, G. and Wiedmann, P. (1994). Using Microworlds to Elaborate more Faithful Models for Reasoning in Probability, *International Conference on Teaching Statistics, 2,* (ICOTS 4), Marrakech, July 1994, 323-335.