# THE UNIVERSAL DISTRIBUTION AND MACHINE LEARNING

Ray J. Solomonoff

Visiting Professor, Computer Learning Research Center
Royal Holloway, University of London

IDSIA, Galleria 2, CH–6928 Manno–Lugano, Switzerland
rjsolo@ieee.org http://world.std.com/˜rjs/pubs.html

## 1 Universal Probability Distribution

I will discuss two main topics in this lecture:

First, the Universal Distribution and some of its properties: its accuracy, its incomputability, its subjectivity.

Secondly, I'm going to tell how to use this distribution to create very intelligent machines.

Many years ago - in 1960 - I discovered what we now call the Universal Probability Distribution(Sol 60). It is the probability distribution on all possible output strings of a universal computer with random input. It seemed to solve all kinds of prediction problems and resolve serious difficulties in the foundations of Bayesian Statistics.

Suppose we have a string, $x$, and we want to know its universal probability with respect to machine, $M$. There will be many inputs to Machine $M$ that will give $x$ as output. Say $s_i$ is the $i$th such input. If $s_i$ is of length $L(s_i)$ bits, the probability that a random binary input would be $s_i$ is just $2^{-L(s_i)}$. To get the probability that $x$ will be produced by *any* of its programs, we sum the probabilities of all of them to get $P_M(x)$, the probability assigned to $x$ by the universal distribution, using machine $M$ as reference.

$$P_M(x) = \sum_i 2^{-L(s_i)} \qquad (1)$$

It is easy to use this distribution for prediction: if $x$ is a binary string, then the probability that 1 will be the next symbol of $x$ is just

$$P_M(x1)/(P_M(x0) + P_M(x1))$$

Five years later, in 1965, Kolmogorov, not yet having read my paper, independently discovered "Kolmogorov Complexity". The Kolmogorov Complexity of a string of symbols, $x$, is the length of the shortest program for a reference universal computer that produces $x$ as output. It is closely related to the Universal Distribution. If $K$ is the Kolmogorov Complexity of $x$ then $2^{-K}$ is an approximation to the probability of $x$ obtained by the universal distribution. This is easy to see, since the shortest program for $x$ will give the most weight of all of the terms in equation 1.

Initially Kolmogorov was interested in the mathematical properties of this complexity – in particular, he used it to define randomness. He defined $x$ to be random if its Kolmogorov Complexity is about the same as the length of $x$ in bits. He was surprised to learn of my earlier work on inductive inference and publicized my discoveries in the Soviet Union so, for many years they were much better known there than in the United States.

I was puzzled that Kolmogorov hadn't thought of using these concepts for inductive inference – to define empirical probability — since one of his first great works was the axiomization of probability theory and he had written voluminously on practical applications of probability.

I asked Leonid Levin, who was one of his students at that time, how Kolmogorov could have missed this important discovery. He suggested that inductive inference was at that time, not actually a "mathematical" problem. At first, I wasn't much satisfied with this idea but thinking about it later, it may have been that in 1965 there was no really good definition of induction and certainly no general criterion for how good an inductive system was.

## 2 Inductive Inference and the Convergence Theorem

After my initial discovery I tried to find a criterion for the accuracy of my prediction method and finally thought of a good one: Suppose we have a probabilistic algorithm that can be described in a finite number of bits, and this algorithm produces a long sequence of symbols according to its probabilistic rules. Then suppose we have a general induction system that gives probabilities for each symbol, in terms of the previous symbols. For a good general induction system, and a long enough sequence, the probabilities given to the symbols by the two different methods should be very close.

While this criterion seemed reasonable, I was at first unable to prove that the Universal Distribution satisfied it.

In 1968 I was asked to review a paper on Inductive Inference, by David Willis. Though I was familiar with the ideas in the paper, it took me about 6 months to really understand it.

Willis had taken my system for induction and made it into an exact, rigorously defined system. He had an error criterion it satisfied, but it was certainly not enough to convince anyone that the system was good for prediction. He showed that the *average* ratio of the correct probability to the estimated probability approached one as the length of data sequence increased — the individual probability ratios could, however, be quite large or quite small. The individual true and estimated probabilities could be quite different.

However, I was able to improve this result to show that the expected values of the sum of the squares of the differences in probabilities between the correct and the estimated values was bounded by a constant. The errors in the individual bit probability values had to approach zero faster than 1/n , n being the length of the sequence. This was a very powerful result.

I called it *The Convergence Theorem*

This theorem made it clear that the universal distribution gave *very good* probability estimates.

I sent in a strong recommendation that Willis' paper be published with no revisions — but the other two reviewers had already rejected it — they felt that it added little to my original paper!

I wrote Willis telling him what a great paper is was and suggested that he send it to another journal. He did this, and it was published two years later.

The first Convergence Theorem was for the Normalized Universal Distribution on potentially infinite sequences of symbols (Sol 78). Peter Gács (Gac 97) showed that it was also true for an Unnormalized Semimeasure. Then Marcus Hutter (Hut 02) showed it worked for an arbitrary (not-binary) alphabet and for a variety of Loss Functions - one of them very general.

More recently, I showed that it's true for Grammatical Induction - in which the data is a set of finite strings. It also works for Operator Induction in which these finite strings are probabilistic answers to questions that have been generated by an unknown stochastic question answering algorithm(Sol 02).

While the accuracy of the universal distribution as a predictor was certainly critical, other important features were discovered:

- The data need not be stationary: subsequences of data can be missing: the data can be multidimensional — extending finitely or infinitely in all positive and/or negative directions.

- It is often possible to obtain predictions using a truly a priori probability distribution (obtained before the data was known). Under these conditions there is no underfitting or overfitting — the data need not be divided into *trainingset* and *testset.* — All data can be used for testing.

- It is possible to use partial recursive functions to model the data. To my knowledge no one has actually tried this, but the system I will describe later, *will* do it. Whether it gets better results than using only recursive models, remains to be seen!

## 3 Incomputability of Universal Probability

While these features are all very beautiful, there seemed at first to be a quite serious problem — that universal probability was incomputable. Surprisingly enough, this turned out to be not a *Bug* but a *Feature*!

Let me explain: Many years ago in ancient Greece, the Pythagoreans discovered that $\sqrt{2}$ could not be expressed as the ratio of two integers. It took the mathematical community many centuries to get a good understanding of this problem, but well before that time, approximations were made and used. None of the approximations were actually $\sqrt{2}$, but they got arbitrarily close.

In the case of the Universal Distribution, we can make a sequence of approximations and just as for $\sqrt{2}$, the approximations will eventually get arbitrarily close to the Platonic ideal — the true Universal Distribution. The difference in the two situations is that for each approximation to $\sqrt{2}$, , we know a good upper bound on how large the error is. On the other hand, for our approximations to the Universal Distribution, we cannot ever know a useful upper bound on how much we deviate from the ideal Distribution.

Fortunately, for almost all applications, we don't need this information. What we usually want to know, is not now close our approximation is to the ideal, but rather, how accurate is our approximation for prediction. If we have a reasonable sample size we can estimate this accuracy by cross validation, but often we can do better. If the approximation we use is entirely a priori (devised before the data was known), we can use all of the data for testing, since none is needed for training the model.

Though the incomputability of the Universal Distribution is usually not relevant to problems in practical prediction, it is of much interest in the Philosophy of Science.

Many scientists are repeatedly disturbed by the need to revise their understanding of their sciences. They look forward to a "Final Theory" that will put an end to all revisions. However, the incomputability of the Universal Distribution assures us that this cannot ever happen. With finite computing resources, we can never be certain that we've found the best, the "Final Theory".

I, personally, am not at all disturbed by this state of affairs, but find it instead to be a never ending source of joy in discovery!

# 4 Prior Information

Another apparent difficulty with the Universal Distribution is its subjectivity. When the Universal Distribution is mentioned, there are two possible meanings of the term "Universal": First, that the error will converge to zero rapidly if the algorithm generating the data has a small finite description. — This is true for all such generating algorithms.

This is what I mean by Universal Distribution.

Another interpretation of Universality is that we can usefully employ the same Universal Distribution for all problems. This is what is called a *half truth*. The same Universal Distribution will indeed work for all problems, but for most, it will work poorly — the errors will converge very slowly. To get good predictions it is usually necessary to use a different Universal Distribution for each Problem Domain. Choice of the distribution will depend on a priori information – the information known $before$ the statistician sees the data.

As soon as data is used to solve a problem, the statisticians a priori is *updated* to reflect that solution — so we have a continually changing a priori probability distribution throughout the life of the statistician that reflects the problems solved during his or her life.

A philosopher may ask: is there not a universal a priori probability distribution in which you have *no* prior information?

To answer this question, let me give an analogy:

If I had no food, water or air to breathe, what would I do? — very little to be done — I would die quickly .

Similarly, if I had no a priori information, there would be little that I could do to solve a statistical problem (or any other problem for that matter!).

Fortunately, we don't ever get into this situation: we are born with fairly good a priori knowledge of the world we are to live in. This a priori information enables us to learn to walk, to learn to communicate and to learn to adapt to hostile environments. It is very unlikely that we would ever learn these things, if we didn't have this a priori information. The exact nature of the a priori information that a person has, is difficult to characterize. However, we normally have to use only part it. For a specific problem, we often have strong ideas as to what functions would be useful to solve it – in which case, we would augment the instruction set of our universal computer with those functions. If we are less certain of what functions are needed, we might use a set of instructions that has been designed for a more general prediction method — such sets of instructions are in C++ libraries, or parts of Mathematica, Maple or Matlab. If the instructions inserted are not relevant to the correct probability function, the convergence rate will be slow, but it will converge eventually to the correct values.

This subjectivity — the fact that they are based on choice of which Universal machine to use — is characteristic of all prediction systems based on a priori probability distributions. The choice of Universal machine and its instruction set is a necessary parameter in the system that enables us to insert a priori information into it.

The dependence of the universal distribution on choice of machine is not a *Bug* in the System — it, too, is a *Necessary Feature.*

# 5 Intelligent Machines

My main goal in studying universal distributions was not especially prediction, but strong AI – for me, this meant writing a program that could work most scientific problems much better than humans can.

Many years ago – about the time of the discovery of the Universal Distribution — Newell and Simon programmed GPS — General Problem Solver
(New 61)— a program that was meant to solve a great variety of problems.

In fact, it only solved a small subset of what we call "Inversion problems" in a very deterministic way. Perhaps its most important defects were that it had no concept of probability and it was absolutely unable to learn.

Suppose you gave it a problem and after a long time, with great difficulty, it finally solved it . If you gave it the same problem the next day , it would solve the problem in the same difficult way, taking the same length of time.

Nevertheless, the A.I. community was pretty much taken with GPS and the Expert Systems that followed and for many years there was relatively little work in A.I. involving learning or probability.

About 1984, roughly 25 years later, at an annual meeting of the American Association for Artificial Intelligence (AAAI), a vote was taken and it was decided that probability was in no way relevant to Artificial Intelligence.

A protest group quickly formed and the next year there was a workshop at the AAAI meeting devoted to Probability and Uncertainty in A.I. This workshop has continued to the present day to be a yearly event.

As part of the protest at the first workshop, I gave a paper on applying the universal distribution to problems in A.I.(Sol 86) This was an early version of the system that I've been developing since that time (Sol 89, 02) .

My interest has always been in a much more general class of problem solver than that originally envisioned by Newell and Simon. The system I've been working on solves problems with both probabilistic and deterministic answers and learning is an integral part of the system.

It is designed to learn to solve two kinds of problems. Almost all problems in science and engineering are of these two kinds.

The first kind is function inversion. These are the P and NP problems of computational complexity theory. They include theorem proving, solution of equations, symbolic integration, etc.

The second kind of problem is time limited optimization. Inductive inference, surface reconstruction, and image restoration are a few examples of this kind of problem.. Designing an automobile in 6 months satisfying certain specifications and having minimal cost, is another.

The general understanding of probability that we have obtained through the universal distribution, has enabled us to design programs that can learn to solve both of these kinds of problems in a manner that seems to follow the acquisition of new skills by humans.

In the infant machine, we have a set of problem solving techniques inserted by the trainer. We have a conditional probability distribution based on the previous experience of the trainer and the machine, that suggests which problem solving techniques should be used with which problems.

The experienced system has many more problem solving techniques. When the system is given a new problem it uses its previous experience with similar problems to decide which problem solving techniques to try and how much time to spend on each trial. This experience is embodied in a General Conditional Probability Distribution. This distribution gives the probability that each problem solving technique will be the best technique for solving any particular problem. The *condition* on the probability distribution is the problem to be solved, and the distribution itself will be on the probability that each problem solving technique will be the best way to solve that particular problem.

The system uses this probability distribution to decide how much time to spend on each problem solving technique. After the problem is solved the General Conditional Probability Distribution is modified, the problem solving techniques may be modified, augmented or deleted in view of this recent experience.

# 6 Concluding Remarks

The last talk I gave at Royal Holloway was at a symposium on the "Importance of Being Learnable." I discussed some approaches to "transfer learning" – how learning in one domain could utilize information from other apparently disparate domains.

Much of my work of recent years has been in developing and understanding the updating system that enables the General Conditional Probability Distribution to implement direct learning and transfer learning from both successful and unsuccessful problem solving trials (Sol 02)

Recently, Juergen Schmidhuber has programmed OOPS(Sch 02), a system for incremental learning, inspired in part by my work in this area.

# References

[1] (Hut 02) Hutter, M.,"Optimality of Universal Bayesian Sequence Prediction for General Loss and Alphabet," http://www.idsia.ch/˜marcus/ai/

[2] (Gac 97) Gács, Peter, "Theorem 5.2.1," in *An Introduction to Kolmogorov Complexity and Its Applications* , Li, M. and Vitányi, P. , Springer-Verlag, N.Y., 1997, pp. 328-331

[3] (New 61) Newell, A. and Simon, H.A., "GPS, a Program That Simulates Human Thought," in *Computers and Thought*, eds, Feigenbaum and Feldman, McGraw-Hill, N.Y., 1963.

[4] (Sch 02) Schmidhuber, J., "Optimal Ordered Problem Solver," TR IDSIA-12-02, 31 July 2002. http://www.idsia.ch/˜juergen/oops.html

[5] (Sol 60) Solomonoff, R.J. "A Preliminary Report on a General Theory of Inductive Inference," (Revision of Report V–131), Contract AF 49(639)–376, Report ZTB–138, Zator Co., Cambridge, Mass., Nov, 1960. http://world.std.com/˜rjs/pubs.html

[6] (Sol 78) Solomonoff, R.J., "Complexity–Based Induction Systems: Comparisons and Convergence Theorems," *IEEE Trans. on Information Theory*, Vol IT–24, No. 4, pp. 422–432, July 1978. http://world.std.com/˜rjs/pubs.html

[7] (Sol 86) Solomonoff, R.J. "The Application of Algorithmic Probability to Problems in Artificial Intelligence," in *Uncertainty in Artificial Intelligence*,

Kanal, L.N. and Lemmer, J.F. (Eds), Elsevier Science Publishers B.V., 1986, pp. 473–491. http://world.std.com/˜rjs/pubs.html

[8] (Sol 89) Solomonoff, R.J. “A System for Incremental Learning Based on Algorithmic Probability,” Proceedings of the Sixth Israeli Conference on Artificial Intelligence, Computer Vision and Pattern Recognition, Dec. 1989, pp. 515–527. http://world.std.com/˜rjs/pubs.html

[9] (Sol 02) Solomonoff, R.J. “Progress in Incremental Machine Learning” Preliminary Report (2002), http://world.std.com/ rjs/nips02.pdf