Most sequences are predictable

V.V. V'yugin

Institute for Information Transmission Problems,
Russian Academy of Sciences,
Bol'shoi Karetnyi per. 19, Moscow GSP-4, 101447, Russia

and

Computer Learning Research Centre
Royal Holloway, University of London
Egham, Surrey TW20 0EX, England

Technical Report

CLRC-TR-99-01

December 1999

Royal Holloway
University of London

Department of Computer Science
Egham, Surrey TW20 0EX, England

Abstract

Recent work in the theory of prediction with expert advice has shown that there exists an intrinsic measure of the difficulty of prediction of data sequences; this measure, which is a generalization of the notion of Kolmogorov complexity, is called "predictive complexity". The predictive complexity of a data sequence gives a limit on the predictive performance of simply described prediction algorithms. We say that a data sequence is "predictable" if there exists a simply described prediction algorithm whose performance is close to the predictive complexity. In this paper we argue that data sequences normally occurring in the real world are predictable; more formally, we prove that Levin's *a priori* semimeasure of non-predictable sequences is small.

1 Introduction

We present the formal results suggesting some possible explanation of the following phenomenon: for many data sets the performance of the best prediction algorithms (Support Vector Machines, neural nets, boosting in the prediction of hand-written digits) is comparable. Possible explanation is based on a new notion of predictive complexity. The predictive complexity of a data sequence gives a lower limit on the predictive performance of simply described prediction algorithms. We suppose that the state-of-the-art algorithms attain predictive complexity.

This is not true for *all* possible data sequences; but we can try to prove this for "normal", in some sense, sequences.

Our problem is closely related to Kolmogorov's theory of stochastic sequences (for the log-loss game), see Shen [4]. Similar results (for stochastic setting) were obtained earlier by V'yugin [9].

The plan of this paper is as follows. First we give formal definitions of predictive strategy, loss function, predictive complexity. After that we discuss the notion of a "normal" sequence (according to Leonid Levin's philosophy) and define the *a priori semimeasure*. In Sections 4 and 5 we state our main results giving lower and upper bounds on the *a priori* semimeasure of non-predictable sequences.

2 Predictive complexity

Suppose we are given a sequence $x_1, x_2, \ldots, x_i \ldots$ of some data. In this paper we consider only the simplest case, where $x_i \in \{0, 1\}$. Our goal is to predict the elements of this data set on-line: we predict x_1, then predict x_2 given x_1, ..., then predict x_{i+1} given $x_1, x_2, \ldots x_i$, etc. At every step i the loss is measured by the some function $\lambda(x_i, p_i)$, where the forecast is a real number p_i and the actual outcome is x_i. We consider only loss functions computable by algorithms. For example, we consider the squared difference $\lambda(x_i, p_i) = (x_i - p_i)^2$ and the log-loss function $\lambda(x_i, p_i) = -\log p_i$ if $x_i = 1$ and $\lambda(x_i, p_i) = -\log(1 - p_i)$ if $x_i = 0$. Here log means logarithm to the base 2. Other loss functions are considered in Vovk [6], Haussler, Kivinen, Warmuth [1] (log-loss, Hellinger, etc.), Vovk and Watkins [8] (financial theory), Yamanishi [13] (logistic, etc).

It is natural to suppose that all predictions are given according to a *prediction strategy* (or *prediction algorithm*) $p_i = S(x_1 x_2 \ldots x_{i-1})$. The total loss incurred by Predictor who follows the strategy S over the first n trials $x_1, x_2, \ldots, x_n$ is defined

$$\mathrm{Loss}_S(x_1 x_2 \ldots x_n) = \sum_{i=1}^{n} \lambda(x_i S(x_1 x_2 \ldots x_{i-1})).$$

The main task is to minimize the total loss suffered on a sequence $x = x_1 x_2 \ldots x_n$ of outcomes. The corresponding game-theoretic interpretation is given in Vovk [7] or in Vovk and Watkins [8].

Let us fix $\eta > 0$ (the *learning rate*) and put $\beta = e^{-\eta} \in (0,1)$. A *generalized prediction* is defined to be any function on finite binary sequences taking real values. Three important classes of generalized predictions g are given by the following definitions from Vovk [6]:

- $\diamond$ g is a *permitted prediction* if there exists a prediction p such that, for all y, $g(y) = \lambda(y,p)$ (i.e., g describes the potential losses implied by p);
- $\diamond$ g is a *superprediction* if there exists a prediction p such that, for all y, $g(y) \geq \lambda(y,p)$ (i.e., g is a permitted prediction, perhaps plus some extra losses);
- $\diamond$ g is an *η-mixture of predictions* if there exist a sequence $(p_1, p_2, \ldots$ of predictions and a sequence $(r_1, r_2, \ldots)$ of nonnegative weights summing to 1 such that, for all y,

$$g(y) = \log_\beta \sum_{i=1}^{\infty} \beta^{\lambda(y,p_i)} r_i.$$

The loss function (game) is *perfectly η-mixable* if every η-mixture of predictions is a superprediction. From this follows, that every η-mixture of superpredictions is also a superprediction. The loss function (game) is *perfectly mixable* if it is η-mixable for some $\eta > 0$. It is known that many popular loss functions such as the log-loss function, square-loss function, Cover's loss function, long-short loss function, Kullback–Leibler loss function, χ^2 loss function, Hellinger loss function etc. (see, e.g., [5], [1], [6] [8] are perfectly mixable.

The important construction in this field is the Vovk's aggregating algorithm AA [5], [6]. In the case of perfectly mixable loss functions this algorithm given a finite sequence of predictive strategies $S_1, S_2, \ldots S_k$ and weights $r_1, r_2 \ldots r_k$, summing to 1, allows us to define theirs "mixture" - a prediction strategy S such that

$$\mathrm{Loss}_S(x) \leq \log_\beta \sum_{i=1}^{k} r_i \beta^{\mathrm{Loss}_{S_i}(x)} \tag{1}$$

for all x, where $\beta = e^{-\eta}$ and the corresponding loss function is η-mixable. The exact construction is given in Section A.

Later Vovk extended the idea of AA to arbitrary infinite sequence of loss processes. In the framework of Kolmogorov complexity for a wide class of computable loss functions he defined a minimal possible total loss of prediction. Now we briefly review the concept of *predictive complexity* from Vovk and Gammerman [7] and Vovk and Watkins [8] in binary case.

Without loss of generality we consider the loss processes corresponding to computable prediction strategies S. In this case, the value $\mathrm{Loss}_S(x)$ can be interpreted as predictive complexity of x. This value, however, depends on S and it is unclear which S to choose. In the most interesting cases a smallest loss functions does not exist - given a computable prediction strategy S, it is easy to construct a computable prediction strategy that greatly outperforms S on at least one outcome sequence. Levin [12], developing ideas of Kolmogorov and Solomonoff, suggested (for a particular loss function) a very natural solution to the problem of nonexistence of a smallest computable loss process. Vovk [7] extended these ideas in a more general setting - for arbitrary loss-prosesses.

We will say that a function $K(x)$ is a *measure of predictive complexity* if the following two conditions hold:

- ◇ (i) K is a *superloss process*, which means that $K(\Box) = 0$ (where $\Box$ is the empty sequence) and for any x there exists an p such that

$$K(xj) \geq K(x) + \lambda(j, p) \tag{2}$$

  for each $j = 0, 1$.

- ◇ (ii) K is *semicomputable from above*, which means that there exists a computable sequence of computable functions K^t such that, for every x, $K(x) = \inf_t K^t(x)$.

Requirement (i) means that the measure of predictive complexity must be valid: there must exists a prediction strategy that achieves it. (Notice that if $\geq$ is replaced by $=$ in (2), the definition of a loss process will be obtained.) Requirement (ii) means that it must be "computable in the limit"; since we are interested in a universal measure of predictive complexity K, we cannot hope that we will be able to compute it in finite time; all we can do is to output more and more accurate approximations K_i to it so that in the limit we obtain K. (In (ii) we can assume, without loss of generality, that the sequence K_n is decreasing.)

A smallest, to within an additive constant, measure K of predictive complexity will be said to be *universal*. In other words, a measure K of predictive complexity is universal if, for any other measure K' of predictive complexity,

$$K(x) \leq K'(x) + c,$$

for some constant c (depending on K'). In Vovk and Gammerman [7] and Vovk and Watkins [8] it was shown that for any perfectly mixable loss function (game) a universal measure of predictive complexity $KG(x)$ can be defined

$$KG(x) = \log_\beta \sum_{i=1}^{\infty} \beta^{K_i(x)} r_i, \tag{3}$$

where $K_i(x)$ is semicomputable from above sequence of all measures of predictive complexity, r_i be a semicomputable from below sequence of real numbers such that the the series $\sum_{i=1}^{\infty} r_i$ is convergent and its sum does not exceed 1. For instance, we can take $r_i = 2^{-K(i)}$, where $K(i)$ is the prefix Kolmogorov complexity of a program i (see e.g. Li and Vitanyi [3], Section 3). The conditions (i) and (ii) for $KG(x)$ will be verifyed in Section A.

$KG(x)$ will be called the *predictive complexity* of x in G.

By (3) for any computable prediction strategy S it holds

$$KG(x) \leq \mathrm{Loss}_S(x) + c_\eta K(S) + c, \tag{4}$$

where $c_\eta = (\eta \log e)^{-1}$, c is a positive constant and $K(S)$ is the Kolmogorov prefix complexity of the program computing S.

3 A priori semimeasure

The concept of predictive complexity is based on thoughtful ideas of Solomonoff and Kolmogorov on the existence of the universal objects in some classes of constructive objects. Solomonoff proposed the philosophical ideas on defining of the *a priori* probability distribution on the basis of the general theory of algorithms. The main problem was that the maximal computable probability distribution does not exist.

Levin [12] gives a precise form of Solomonoff ideas in a concept of a maximal semimeasure semicomputable from below (see also [3], Section 4.5). A real-valued function $P(x)$, where x is a finite binary sequence, is called semimeasure if

- ◇ $P(\Box) \leq 1$;
- ◇ $P(x) \geq P(x0) + P(x1)$ for all x;
- ◇ the function P is semicomputable from below (see (ii) of the analogous definition of predictive complexity).

Levin proved that there exists a maximal to within a multiplicative constant factor semimeasure M semicomputable from below, i.e. for any semimeasure P semicomputable from below a constant c exists such that the inequality

$$cM(x) \geq P(x) \tag{5}$$

holds for all x.

It is easy to see that the function $KL(x) = -\log M(x)$ is a variant of the predictive complexity for log-loss function (Levin game). ¿From (4) it follows that log-loss complexity is the maximal (to within a constant factor) among predictive complexities from a very natural class.

Proposition 1 *For any η-mixable loss function $\lambda(\omega, p)$ (game G) satisfying $\lambda(0,0) = \lambda(1,1) = 0$ and any $\delta > 0$ a constant c exists such that the following hold*

- ◇ *(i) $KG(x) \leq c_\eta K(x) + c$;*

$\diamond$ *(ii)* $KG(x) \leq (1+\delta)c_\eta KL(x) + c$,

where $c_\eta = (\eta \log e)^{-1}$.

Proof. To prove the item (i) consider the prediction strategy S which on any sequence of the length $i-1$ outputs the i-th element x_i of the sequence $x = x_1 \ldots x_n$ using the code of x under an optimal mode of description of finite objects. The length of this code is equal to the prefix complexity $K(x)$ [3], Section 3. By definition $\mathrm{Loss}_S(x) = 0$. The item (ii) follows from the inequalities between $K(x)$ and $KL(x)$ [3], Section 4. $\square$

At the rest of this section we consider some notions needed to give an interpretation of Theorem 2 below. Levin [2] considered the combinations of probabilistic and deterministic processes as the most general class of processes of generating data. With each probabilistic process some computable probability distribution can be assigned. Each deterministic process is realized by means of an algorithm. Algorithmic processes transformed sequences generating by probabilistic processes into new sequences. More precise, a probabilistic computer F is a Turing machine supplied with an additional input tape. In the process of computation this machine reads a sequence ω on this tape and produces a sequence $\omega' = F(\omega)$. We suppose that there is a computable probability distribution μ in the set of all possible ω. So we can calculate the probability

$$P(x) = \mu\{\omega | x \subseteq F(\omega)\}$$

of that the result $F(\omega)$ of the computation begins with a finite sequence x. More precise, $P(x)$ is not a probability distribution, since $F(\omega)$ may be finite for an infinite ω. It is easy to see that $P(x)$ is a semimeasure semicomputable from below. The converse result is proved in [12]: for any semimeasure $P(x)$ semicomputable from below a probabilistic computer $F(\omega)$ exists such that

$$P(x) = \mu\{\omega | x \subseteq F(\omega)\},$$

for all x, where $\mu(x) = 2^{-l(x)}$ is the uniform probability distribution in the set of all binary sequences.

Therefore, by (5) $M(x)$ is an universal upper bound of the probability of generating sequences x by probabilistic computers.

Let A be any set of the binary sequences of the length n. We define

$$M(A) = \sum_{x \in A} M(x).$$

Let some property Π_n defines for any n a set A_n of binary sequences of the length n. Then for any probability distribution μ and algorithmic process F there exists a constant c (depending only on μ and F) such that

$$\mu\{\omega | F(\omega) \in A_n\} \leq cM(A_n)$$

for all n. According to our framework, if $M(A_n) \longrightarrow 0$ as n tends to infinity, then for all sufficiently large n the sequences of the length n having property Π_n form a scarce part of all objects generating in combinations of deterministic and probabilistic processes.

4 Predictable and non-predictable sequences

Let α and γ be some nonnegative numbers. A sequence x of the length n is called (α, γ)-*predictable* if there exists a prediction strategy S such that $K(S) \leq \alpha$ and

$$\mathrm{Loss}_S(x) - KG(x) \leq \gamma.$$

The interesting case when resource bound α is small with respect to the length n of the sequence x (for instance $\alpha = O(\log n)$).

The following Theorem 1 shows that for a wide class of loss functions non-predictible sequences exist. Consider some conditions for a loss function sufficient to following theorem holds:

- ⋄ (i) $\lambda(0,0) = \lambda(1,1) = 0$;
- ⋄ (ii) there exists a computable positive number b such that $\lambda(0,p) \geq b$ or $\lambda(1,p) \geq b$ for each p;
- ⋄ (iii) the loss function $\lambda(\omega, p)$ is perfectly mixable.

The log-loss function and the squared difference satisfy these condition with $b = 1$, and $b = \frac{1}{4}$, accordingly.

Theorem 1 *For any loss function satisfying the conditions (i)-(iii) above a positive constant c exists such that for any n there exists a binary sequence x of the length n satisfying*

- ⋄ $\mathrm{Loss}_P(x) - KG(x) \geq bn - 2c_\eta\alpha - c_\eta(\log n + 2\log\log n) - c$ *for each prediction strategy P such that $K(P) \leq \alpha$, where $c_\eta = (\eta \log e)^{-1}$;*
- ⋄ $M(x) \geq 2^{-\alpha - \log n - 2\log\log n - c}$.

5 Most sequences are predictable

The predictive complexity determines asymptotically the minimal possible loss of forecasting. It includes also arbitrarily complex prediction strategies. Here we impose the restriction $K(S) \leq \alpha$, where α reflecting a degree of computational resources allowed. We show that even in the case when α is small with respect to the length n of binary sequences (for instance $\alpha = O(\log n)$) it can help us to reach almost the minimal possible loss incurred over the most elements of the data set.

In this section we estimate how large can be the set of non-predictable sequences of the length n. We prove that the a priori semimeasure of this set is asymptotically decreases as n increases.

Let $D^n_{\alpha,\gamma}$ be the set of all binary sequences of the length n which are not (α,γ)-predictable. For any $x \in D^n_{\alpha,\gamma}$ we have

$$\mathrm{Loss}_S(x) - KG(x) > \gamma$$

for each prediction strategy S such that $K(S) \le \alpha$.

Theorem 2 ◇ *For any perfectly mixable loss function satisfying condition (i) from Section 4 a constant c exists such that, for any n, α and $1 \le \gamma \le n$ the estimate*

$$M(D^n_{\alpha,\gamma}) = \sum_{x\in D^n_{\alpha,\gamma}} M(x) \le 2^{-\alpha+2\log n+2\log\log n-\log\gamma+c} \tag{6}$$

holds;

◇ *If the predictive complexity is based on the log-loss function the corresponding estimate is the following*

$$M(D^n_{\alpha,\gamma}) = \sum_{x\in D^n_{\alpha,\gamma}} M(x) \le 2^{-\alpha+\log n+2\log\log n+c}, \tag{7}$$

This theorem is a generalization of Theorem 3 [9] for wide class of loss functions.

Inequality (6) or (7) can be interpreted as an upper bound on the probability of generating non-predictible sequences by a probabilistic computer.

For any m by Theorem 2 the probability of generating (α,γ)-*predictable* sequences is $\ge 1-2^{-m}$ when $\alpha+\log\gamma > 2\log n+\log\log n+c+m$. The last condition wittingly holds for any $1\le\gamma\le n$ if

$$\alpha \ge 2\log n + \log\log n + c + m. \tag{8}$$

Theorem 2 shows that given an upper bound α satisfying (8) on the complexity of prediction strategies most sequences x are predictable with respect to the simple prediction strategy Q with $K(Q)\le\alpha$, i.e. $\mathrm{Loss}_Q(x)$ is close to its minimal value $KG(x)$.

By Theorem 1 non-learnable objects can also exist, but by Theorem 2 and Levin's philosophy we shall meet them very rarely; non-predictable fluctuations of prices on financial markets formed a scarce part in the stream of all financial data.

Now we formulate the main corollary from Theorems 1 and 2.

Corollary 1 *For any loss function satisfying conditions (i)-(iii) of Section 4 a positive constant c exists such that*

$$2^{-\alpha-\log n-2\log\log n-c} \le M(D^n_{\alpha,\gamma}) \le 2^{-\alpha+2\log n+2\log\log n-\log\gamma+c}$$

for each $0\le\gamma\le bn-2c_\eta\alpha-c_\eta(\log n+2\log\log n)-c$.

The logarithmic additive terms in the exponent of (7) can be eliminated (and $\log n + 2\log\log n$ can be subtracted in the exponent of (6)) if we replace the condition $K(S) \le n$ on $K(S|n) \le n$ in the definition of $D^n_{\alpha,\gamma}$ (this follows from the proof of Theorem 2). An open problem arises - can we eliminate or minimize the remainder term $\log n$ in the right-side of the inequality (6).

6 Proofs

6.1 Proof of Theorem 1

For any α let $P_1, P_2, \ldots, P_k$ be all computable prediction strategies with complexity $\le \alpha$. Evidently, $k < 2^{\alpha+1}$. By means of AA we can define a mixture P_α of these strategies such that

$$\mathrm{Loss}_{P_\alpha}(x) \le \log_\beta \sum_{i=1}^{k} k^{-1} \beta^{\mathrm{Loss}_{P_i}(x)}. \tag{9}$$

By construction of AA we can compute k and all values of $P_i(z), l(z) \le n$, with degree of accuracy given below by means of program p with the following properties. The length of p is $\le \alpha$ and this program has the maximal terminating time among all programs computing values $P_j(z)$ for all $j \le k$ and all z of the length $\le n$ with given corresponding degree of accuracy.

After that we can define a sequence x as follows

$$x_s = \begin{cases} 1, \text{ if } \lambda(1, P_\alpha(x_1 \ldots x_{s-1})) > b - 2^{-(s+1)}, \\ 0, \text{ if } \lambda(0, P_\alpha(x_1 \ldots x_{s-1})) > b - 2^{-(s+1)}. \end{cases}$$

for all $1 \le s \le n$, where $x_0 = \Box$. Since these conditions are overlapped, we choose that was first defined. As follows from the definition $\mathrm{Loss}_{P_\alpha}(x) \ge bn - 1$. By (9)

$$\mathrm{Loss}_{P_i}(x) \ge bn - c_\eta \alpha - 1$$

for all $i \le k$ (i.e. for all P such that $K(P) \le \alpha$), where $c_\eta = (\eta \log e)^{-1}$.

By definition $K(x|n) \le \alpha + c$, for some positive constant c. The ordinary inequalities between conditional and unconditional prefix complexities (see [3], Section 3) implies

$$K(x) \le K(x|n) + \log n + 2\log\log n + c \tag{10}$$

for some positive constant c. Adding (i) of Proposition 1 we obtain

$$KG(x) \le c_\eta(\alpha + \log n + 2\log\log n) + c$$

for some c. Hence, for all P such that $K(P) \le \alpha$, we have

$$\mathrm{Loss}_P(x) - KG(x) \ge bn - 2c_\eta\alpha - c_\eta(\log n + 2\log\log n) - c$$

for some positive constant c.

Since $KL(x) \leq K(x) + c$ holds for some c (see [3], Section 4.5), by (10) we have

$$M(x) \geq 2^{-\alpha - \log n - 2 \log\log n - c}.$$

$\square$.

6.2 Proof of Theorem 2

Simple intuitive explanation of the idea of the proof: the α-simple prediction strategy will be an approximation to the universal "prediction semistrategy".

Since the universal measure of predictive complexity is semicomputable from above, the function $Q(x) = \beta^{KG(x)}$ is lower semicomputable. This means that there exists a non-decreasing computable sequence of functions $k_i(x)$ taking rational values such that $Q(x) = \sup_i k_i(x)$.

The proof is much simpler when $KG(x) = KL(x) = -\log M(x)$ is the Levin's predictive complexity for log-loss function. In this case $\beta = \frac{1}{2}$ and $Q(x) = M(x)$. By definition of M we have

$$\sum_{l(x)=n} \beta^{KG(x)}(x) = \sum_{l(x)=n} M(x) \leq 1. \tag{11}$$

Let p be a finite binary sequence representing the rational approximation of the real number $\sum\limits_{l(x)=n} M(x)$ from below with accuracy $2^{-\alpha}$. Then using p and n we can effectively find integer numbers t and k such that the following conditions hold

- ⋄ (i) $\sum\limits_{l(x)=n} \beta^{KG^{t,k}(x)} > \sum\limits_{l(x)=n} \beta^{KG(x)} - 2^{-\alpha}$, where $KG^{t,k}(x) = \log_\beta \sum\limits_{i=1}^{k} \beta^{K_i^t(x)} 2^{-i}$, and $K_i^t(x)$ is some approximation from above of $K_i(x)$ computed in t steps.
- ⋄ (ii) for each x of the length $\leq n$ and for each $i \leq k$ the difference $K_i^t(xj) - K_i^t(x)$ is a superprediction.

The mixture of superpredictions is also a superprediction. Then by (ii) the difference $KG^{t,k}(xj) - KG^{t,k}(x)$ is a superprediction for each x of the length $\leq n$, and by the definition of the superprediction and by AA there exists a prediction strategy Q such that $\mathrm{Loss}_Q(x) \leq KG^{t,k}(x)$ for all such x. Since this construction is algorithmically effective,

$$K(Q|n) \leq \alpha + c, \tag{12}$$

where c is a constant.

Temporarily in the definition of $D^n_{\alpha,\gamma}$ we will consider prediction strategies S α-simple conditional with respect to n, i.e. such that $K(S|n) \leq \alpha$.

By definition for any $x \in D^n_{\alpha,\gamma}$ we have

$$\mathrm{Loss}_S(x) - KG(x) > \gamma$$

for each prediction strategy S such that $K(S|n) \leq \alpha$.

Therefore, by (3) and (i) for any $x \in D^n_{\alpha-c,\gamma}$ we obtain

$$\beta^\gamma \sum_{x \in D^n_{\alpha-c,\gamma}} \beta^{KG(x)} > \sum_{x \in D^n_{\alpha-c,\gamma}} \beta^{\mathrm{Loss}_Q(x)} \geq$$
$$\sum_{x \in D^n_{\alpha-c,\gamma}} \beta^{KG^{k,t}(x)} > \sum_{x \in D^n_{\alpha-c,\gamma}} \beta^{KG(x)} - 2^{-\alpha},$$

where c is such that (12) holds. From this it follows

$$(1-\beta^\gamma) \sum_{x \in D^n_{\alpha-c,\gamma}} \beta^{KG(x)} \leq 2^{-\alpha}. \tag{13}$$

In the case of log-loss function $\beta = \frac{1}{2}$ and $\beta^{KG(x)} = M(x)$, and then by (13) we have

$$M(D^n_{\alpha-c,\gamma}) \leq 2^{-\alpha+1}$$

for any $\gamma \geq 1$.

For other types of predictive complexities the sum $\sum_{l(x)=n} \beta^{KG(x)}$ can exceed 1. For instance, in the case of square-loss function this sum is of order of exponent from n. Let us consider the general case. We replace the inequality (11) on

$$\sum_{l(x)=n} \beta_n^{KG(x)} M(x) \leq 1,$$

where $\beta_n = e^{-\frac{1}{n}}$. Let p be a finite binary sequence representing the rational approximation of the real number $\sum_{l(x)=n} \beta_n^{KG(x)} M(x)$ from below with accuracy $2^{-\alpha}$. After that, as above using p and n we effectively find t and k such that the conditions (i') and (ii) hold, where the condition

- $\diamond$ (i') $\sum_{l(x)=n} \beta_n^{KG^{t,k}(x)} M^t(x) > \sum_{l(x)=n} \beta_n^{KG(x)} M(x) - 2^{-\alpha}$.

is used instead of condition (i) above. Here

$$KG^{t,k}(x) = \log_\beta \sum_{i=1}^k \beta^{K_i^t(x)} 2^{-i},$$

where $\beta = e^{-\eta}$ and η is such that our loss function is η-mixable, $K_i^t(x)$ is some approximation from above of $K_i(x)$ and $M^t(x)$ is some approximation from below of $M(x)$ computed in t steps.

There exists a prediction strategy Q such that $\mathrm{Loss}_Q(x) \le KG^{t,k}(x)$ for all x of the length $\le n$ and $K(Q|n) \le \alpha + c$, where c is a constant. ¿From this and (i') we obtain

$$\sum_{l(x)=n} \beta_n^{\mathrm{Loss}_Q(x)} M(x) \ge \sum_{l(x)=n} \beta_n^{KG(x)} M(x) - 2^{-\alpha}. \qquad (14)$$

By definition for any $x \in D^n_{\alpha-c,\gamma}$ we have

$$\mathrm{Loss}_Q(x) - KG(x) > \gamma.$$

Therefore, by (14) we obtain

$$\beta_n^\gamma \sum_{x\in D^n_{\alpha-c,\gamma}} \beta_n^{KG(x)} M(x) > \sum_{x\in D^n_{\alpha-c,\gamma}} \beta_n^{\mathrm{Loss}_Q(x)} M(x) \ge \sum_{x\in D^n_{\alpha-c,\gamma}} \beta_n^{KG(x)} M(x) - 2^{-\alpha}.$$

¿From this it follows

$$(1-\beta_n^\gamma) \sum_{x\in D^n_{\alpha-c,\gamma}} \beta_n^{KG(x)} M(x) \le 2^{-\alpha}. \qquad (15)$$

By condition (ii) of Proposition 1 for any $\delta > 0$ a constant $c > 0$ exists such that $KG(x) \le (1+\delta)c_\eta KL(x) + c$. We have also $KL(x) \le n + c$ for some $c > 0$ ([3], Section 4.5). Hence $KG(x) \le cn$ for some $c > 0$, where n is the length of x.

Since $\beta_n = e^{-\frac{1}{n}}$ we have

$$\beta_n^{KG(x)} = e^{-\frac{1}{n}KG(x)} \ge e^{-c},$$

and

$$1 - \beta_n^\gamma \ge \frac{\gamma}{2n}$$

for $0 < \gamma \le n$. ¿From this and (15) we obtain the estimate

$$\sum_{x\in D^n_{\alpha-c,\gamma}} M(x) \le 2^{-\alpha} \frac{2n}{\gamma} e^c = 2^{-\alpha+\log n - \log\gamma + c\log e + 1}. \qquad (16)$$

To eliminate the condition n in $K(Q|n)$ we consider the following estimate of prefix Kolmogorov complexity [3], Section 3.

$$K(Q) \le K(Q|n) + K(n) + c' \le \alpha + \log n + 2\log\log n + c$$

for some positive constants c' and c. Replacing α in (16) on $\alpha - \log n - 2\log\log n - c$ and returning to the previous (unconditional) definition of $D^n_{\alpha,\gamma}$ we obtain the needed estimate

$$\sum_{x\in D^n_{\alpha,\gamma}} M(x) \le 2^{-\alpha+2\log n+2\log\log n-\log\gamma+c} \qquad (17)$$

for some positive constant c. $\square$

Acknowledgements

The motivation of this paper is due to Volodya Vovk and Alex Gammerman. Author is deeply grateful to them for valuable discussions.

A Appendix

A.1 Aggregating algorithm

In this section we present the Vovk's aggregating algorithm AA. Let a loss function $\lambda(\omega, p)$ be η-mixable and $\beta = e^{-\eta}$. Let also a finite sequence $P_1, P_2, \ldots, P_k$ of computable prediction strategies and a sequence $r(1), \ldots, r(k)$ of nonnegative real numbers summing to 1 are given.

A computable prediction strategy P will be defined such that for any binary sequence $y_1 \ldots y_n$ the inequality

$$\mathrm{Loss}_P(y_1 \ldots y_m) \le \log_\beta \sum_{i=1}^{k} r(i) \beta^{\mathrm{Loss}_{P_i}(y_1 \ldots y_m)} \tag{18}$$

holds.

Put initial weghts $r_0(i) = r(i), i = 1, \ldots, k$. After each trial $y_j, j = 1, \ldots, n$, the weights are updated as follows

$$r_j(i) = \beta^{\lambda(y_j, P_i(y_1 \ldots y_{j-1}))} r_{j-1}(i).$$

As follows from this definition

$$r_j(i) = \beta^{\mathrm{Loss}_{P_i}(y_1 \ldots y_j)} r_0(i).$$

After each trial y_{j-1} a superprediction g_j is defined

$$g_j(\omega) = \log_\beta \sum_{i=1}^{k} \beta^{\lambda(\omega, P_i(y_1 \ldots y_{j-1}))} r^*_{j-1}(i),$$

where

$$r^*_{j-1}(i) = \frac{r_{j-1}(i)}{\sum_{s=1}^{k} r_{j-1}(s)}$$

are the normalized weights.

Since the loss function is η-mixable for any $j = 1, \ldots, n$ a real number p_j exists (can be effectively computed with arbitrary degree of accuracy) such that

$$\lambda(\omega, p_j) \le g_j(\omega)$$

for all ω $(= 0, 1$ in our case$)$. Define $P(y_1 \ldots y_{j-1}) = p_j$.

By mathematical induction on $t = 1, \ldots n$ we shall prove that

$$\sum_{j=1}^{t} g_j(y_j) = \log_\beta \sum_{i=1}^{k} \beta^{\mathrm{Loss}_{P_i}(y_1 \ldots y_t)} r(i). \tag{19}$$

The inequality (18) will be follow from (19) when $t = n$.

When $t = 1$ we have

$$g_1(y_1) = \log_\beta \sum_{i=1}^{k} \beta^{\lambda(y_1, P_i(\square))} r(i).$$

When $t > 1$ we have

$$\log_\beta \sum_{i=1}^{k} \beta^{\mathrm{Loss}_{P_i}(y_1 \ldots y_t)} r(i) - \log_\beta \sum_{i=1}^{k} \beta^{\mathrm{Loss}_{P_i}(y_1 \ldots y_{t-1})} r(i) =$$
$$\log_\beta \frac{\sum_{i=1}^{k} \beta^{\mathrm{Loss}_{P_i}(y_1 \cdots y_{t-1}) + \lambda(y_t, P_i(y_1 \cdots y_{t-1}))} r(i)}{\sum_{i=1}^{k} \beta^{\mathrm{Loss}_{P_i}(y_1 \cdots y_{t-1})} r(i)} =$$
$$\log_\beta \sum_{i=1}^{k} \beta^{\lambda(y_t, P_i(y_1 \ldots y_{t-1}))} r^*_{t-1}(i) = g_t(y_t).$$

A.2 Properties of predictive complexity

The construction of the predictive complexity $KG(x)$ is based on applying the idea of AA to semicomputable from above sequence $K_i(x)$ of all measures of predictive complexity satisfying conditions (i)-(ii) of Section 2. $KG(x)$ was defined in Section 2 by equation (3), where r_i is a semicomputable from below sequence of real numbers such that the series $\sum_{i=1}^{\infty} r_i$ is convergent and its sum does not exceed 1.

By definition $KG(x)$ is semicomputable from above, i.e (i) holds. We must verify (ii). Indeed, by (3) for any x and $j = 0, 1$

$$KG(xj) - KG(x) = \log_\beta \sum_{i=1}^{\infty} q_i \beta^{K_i(xj) - K_i(x)} \geq$$
$$\log_\beta \sum_{i=1}^{\infty} q_i \beta^{\lambda(j, \gamma_i)} \geq \lambda(j, \gamma),$$

where

$$q_i = \frac{r_i \beta^{K_i(x)}}{\sum_{s=1}^{\infty} r_s \beta^{K_s(x)}}.$$

Here γ_i satisfying

$$K_i(xj) - K_i(x) \geq \lambda(j, \gamma_i)$$

exists since each element of the sequence $K_i(x)$ satisfies the condition (i) of the measure of predictive complexity (Section 2), and γ satisfying the last inequality exists by definition of η-mixability. For further details see [8], Section 7.6.

References

[1] Haussler, D., Kivinen, J. and Warmuth, M.K. (1994) Tight worst-case loss bounds for predicting with expert advice. Technical Report UCSC-CRL-94-36, University of California at Santa Cruz, revised December 1994. Short version in P. Vitányi, editor, *Computational Learning Theory*, Lecture Notes in Computer Science, volume 904, pages 69–83, Springer, Berlin, 1995.

[2] Levin, L.A. (1984) Randoness conservation inequalities; Information and independence in mathematical theories. *Inform. Comp.*, **61**, 15–37.

[3] Li, M. and Vitányi, P. (1997) *An Introduction to Kolmogorov Complexity and Its Applications.* Springer, New York, 2nd edition.

[4] Shen, A. (1983) The concept of (α, β)-stochasticity in the Kolmogorov sense and its properties. *Sov. Math. Dokl.*, 28, 295–299.

[5] Vovk, V. (1990) Aggregating strategies. In M. Fulk and J. Case, editors, *Proceedings of the 3rd Annual Workshop on Computational Learning Theory*, pages 371–383, San Mateo, CA, 1990. Morgan Kaufmann.

[6] Vovk, V. (1998) A game of prediction with expert advice. *J. Comput. Syst. Sci.*, 56:153–173.

[7] Vovk, V. and Gammerman, A. (1999) Complexity estimation principle, *The Computer Journal*, *42:4*, 318–322.

[8] Vovk, V., Watkins, C.J.H.C. (1998) Universal portfolio selection, *Proceedings of the 11th Annual Conference on Computational Learning Theory*, 12–23.

[9] V'yugin, V.V. (1985) Nonstochastic objects, *Problems Inform. Transmission* **21**, 77–88.

[10] V'yugin, V.V. (1987) On the defect of randomness of a finite object with respect to measures with given complexity bounds, *SIAM Theory Probab. Appl.* **32**, 508–512.

[11] V'yugin, V.V. (1998) Non-stochastic infinite and finite sequences, *Theoretical Computer Science.* **207**, 363–382.

[12] Zvonkin, A.K. and Levin, L.A. (1970) The complexity of finite objects and the algorithmic concepts of information and randomness, *Russ. Math. Surv.* **25**, 83–124.

[13] Yamanishi, K. (1995) Randomized approximate aggregating strategies and their applications to prediction and discrimination, in *Proceedings, 8th Annual ACM Conference on Computational Learning Theory*, 83–90, Assoc. Comput. Mach., New York.