Algorithmic Statistics, Prediction and Machine Learning[*]

Alexey Milovanov

Moscow State University, Moscow, Leninskie gory, MSU, Russia
almas239@gmail.com

Abstract

Algorithmic statistics considers the following problem: given a binary string x (e.g., some experimental data), find a "good" explanation of this data. It uses algorithmic information theory to define formally what is a good explanation. In this paper we extend this framework in two directions.

First, the explanations are not only interesting in themselves but also used for prediction: we want to know what kind of data we may reasonably expect in similar situations (repeating the same experiment). We show that some kind of hierarchy can be constructed both in terms of algorithmic statistics and using the notion of a priori probability, and these two approaches turn out to be equivalent (Theorem 5).

Second, a more realistic approach that goes back to machine learning theory, assumes that we have not a single data string x but some set of "positive examples" $x_1, \ldots, x_l$ that all belong to some unknown set A, a property that we want to learn. We want this set A to contain all positive examples and to be as small and simple as possible. We show how algorithmic statistic can be extended to cover this situation (Theorem 11).

1998 ACM Subject Classification H.1.1 Systems and Information Theory

Keywords and phrases algorithmic information theory, minimal description length, prediction, kolmogorov complexity, learning

Digital Object Identifier 10.4230/LIPIcs.STACS.2016.54

1 Introduction and Notation

Let x be a binary string, and let A be a finite set of binary strings containing x. Considering A as an "explanation" (statistical model) for x, we want A to be as simple and small as possible (the smaller A is, the more specific the explanation is). This approach can be made formal in the framework of algorithmic information theory, where the notion of algorithmic (Kolmogorov) complexity of a finite object (a string or a set encoded as a binary string in a natural way) is defined.

The definition and basic properties of Kolmogorov complexity can be found in the textbooks [5], [8], for a short survey see [6]. Informally Kolmogorov complexity of a string x is defined as the minimal length of a program that produces x. This definition depends on the programming language, but there are optimal languages that make the complexity minimal up to a constant; we fix one of them and denote the complexity of x by $C(x)$.

We also use another basic notion of the algorithmic information theory, the *discrete a priory probability*. Consider a probabilistic machine A without input that outputs some binary string and stops. It defines a probability distribution on binary strings: $m_A(x)$ is

[*] This work was partially supported by RaCAF ANR-15-CE40-0016-01 grant and RFBR grant 16-01-00362.

33rd Symposium on Theoretical Aspects of Computer Science (STACS 2016).
Editors: Nicolas Ollinger and Heribert Vollmer; Article No. 54; pp. 54:1–54:13
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

SYMPOSIUM ON THEORETICAL ASPECTS OF COMPUTER SCIENCE

the probability to get x as the output of A. (The sum of $m_A(x)$ over all x can be less than 1 since the machine can also hang.) The functions m_A can be also characterized as lower semicomputable semimeasures (non-negative real-valued functions $m(\cdot)$ on binary strings such that the set of pairs (r, x) where r is a rational number, x is a binary string and $r < m(x)$, is computably enumerable, and $\sum_x m(x) \leqslant 1$). There exists a universal machine U such that m_U is maximal (up to $O(1)$-factor) among all m_A. We fix some U with this property and call $m_U(x)$ the *discrete a priori probability of* x, denoted as $\mathbf{m}\,(x)$. The function $\mathbf{m}$ is closely related to Kolmogorov complexity. Namely, the value $-\log_2 \mathbf{m}\,(x)$ is equal to $C(x)$ with $O(\log C(x))$-precision.

Now we can define two parameters that measure the quality of a finite set A as a model for its element x: the complexity $C(A)$ of A and the binary logarithm $\log |A|$ of its size. The first parameter measures how simple is our explanation; the second one measures how specific it is. We use binary logarithms to get both parameters in the same scale: to specify an element of a set of size N we need $\log N$ bits of information.

There is a trade-off between two parameters. The singleton $A = \{x\}$ is a very specific description, but its complexity may be high. On the other hand, for a n-bit string x the set $A = \mathbb{B}^n$ of all n-bit strings is simple, but it is large. To analyze this trade-off, following [3, 4], let us note that every set A containing x leads to a *two-part description of* x: first we specify A using $C(A)$ bits, and then we specify x by its ordinal number in A, using $\log |A|$ bits. In total we need $C(A) + \log |A|$ bits to specify x (plus logarithmic number of bits to separate two parts of the description). This gives the inequality

$$C(x) \leqslant C(A) + \log |A| + O(\log C(A))$$

(the length of the optimal description, $C(x)$, does not exceed the length of any two-part description). The difference

$$\delta(x, A) = C(A) + \log |A| - C(x)$$

is called *optimality deficiency* of A (as a model for x). As usual in algorithmic statistic, all our statements are made with logarithmic precision (with error tolerance $O(\log n)$ for n-bit strings), so we ignore the logarithmic terms and say that $\delta(x, A)$ is positive and measures the overhead caused by using two-part description based on A instead of the optimal description for x.

Note that this overhead $\delta(x, A)$ is zero for $A = \{x\}$, so the question is whether we can obtain A that is simpler than x but maintains $\delta(x, A)$ reasonably small. This trade-off is reflected by a curve called sometimes that the *profile* of x; this profile can be defined also in terms of randomness deficiency (the notion of (α, β)-stochasticity introduced by Kolmogorov, see [8], [9], [7]), and in terms of time-bounded Kolmogorov complexity (the notion of depth, see [4]).

In our paper we apply these notions to an analysis of the prediction and learning. In Section 2 we consider, for a given string x, all “good” explanations and consider their union. Elements of this union are strings that can be reasonably expected when the experiment that produced x is repeated. We show that this union has another equivalent definition in terms of a priori probability (Theorem 5).

In Subsection 2.5 we consider a situation where we start with several data strings $x_1, \ldots, x_l$ obtained in several independent experiments of the same type. We show that all the basic notions of algorithmic statistics can be extended (with appropriate changes) to this framework, as well as Theorem 5.

2 Prediction Hierarchy

2.1 Algorithmic Prediction

Assume that we have some experimental data represented as a binary string x. We look for a good statistical model for x and find some set A that has small optimality deficiency $\delta(x, A)$. If we believe in this model, we expect only elements from A as outcomes when the same experiment is repeated. The problem, however, is that many different models with small optimality deficiency may exist for a given x, and they may contain different elements. If we want to cover all the possibilities, we need to consider the union of all these sets, so we get the following definition. In the following definition we assume that x is a binary string of length n, and all the sets A also contain only strings of length n.

▶ **Definition 1.** Let $x \in \mathbb{B}^n$ be a binary string and let d be some integer. The union of all finite sets of strings $A \subset \mathbb{B}^n$ such that $x \in A$ and $\delta(x, A) \leqslant d$ is called *algorithmic prediction* d*-neighborhood of* x.

Obviously d-neighborhood increases as d increases. It becomes trivial (contains all n-bit strings) when $d = n$ (then $\mathbb{B}^n$ is one of the sets A in the union).

▶ **Example 2.** If $x = 0 \ldots 0$ (the strings consisting of n zeros), then x' belongs to d-neighborhood of x iff $C(x') \lesssim d$

▶ **Example 3.** If x is a random string of length n (i. e. $C(x) \approx n$) then the d-neighborhood of x contains all strings of length n provided d is greater than some function of order $O(\log n)$.

2.2 Probabilistic Prediction

There is another natural approach to prediction. Since we treat the experiment as a black box (the only thing we know is its outcome x), we assume that the possible models $A \subset \mathbb{B}^n$ are distributed according to their a priori probabilities, and consider the following two-stage process. First, a finite set is selected randomly: a non-empty set A is chosen with probability $\mathbf{m}(A)$ (note that a priori probability can be naturally defined for finite sets via some computable encoding). Second, a random element x of A is chosen uniformly. In this process every string x is chosen with probability

$$\sum_{A \ni x} \mathbf{m}(A)/|A|,$$

and it is easy to see that this probability is equal to $\mathbf{m}(x)$ up to a $O(1)$-factor. Indeed, the formula above defines a lower semicomputable function of x, so it does not exceed $\mathbf{m}(x)$ more than by $O(1)$-factor. On the other hand, if we restrict the sum to the singleton $\{x\}$, we already get $\mathbf{m}(x)$ up to a constant factor. So this process gives nothing new in terms of the final output distribution on the outcomes x. Still the advantage is that we may consider, for a given pair of strings x and y, the conditional probability

$$p(y|x) = \Pr[y \in A \mid \text{the output of the two-stage process is } x].$$

In other words, by definition

$$p(y|x) = \frac{\sum_{A \ni x,y} \mathbf{m}(A)/|A|}{\sum_{A \ni x} \mathbf{m}(A)/|A|}. \tag{1}$$

As we have said, the denominator equals $\mathbf{m}\,(x)$ up to $O(1)$-factor, so

$$p(y|x) = \frac{\sum_{A\ni x,y} \mathbf{m}\,(A)/|A|}{\mathbf{m}\,(x)} \tag{2}$$

up to $O(1)$-factor. Having some string x and some threshold d, we now can consider all strings y such that $p(y|x) \geqslant 2^{-d}$ (we use the logarithmic scale to facilitate the comparison with algorithmic prediction). These strings could be considered as plausible ones to appear when repeating the experiment of unknown nature that once gave x.

Our main result shows that this approach is essentially equivalent to the algorithmic prediction. By a technical reason we have to change slightly the random process that defines $p(y|x)$. Namely, it is strange to consider models that are much more complex than x itself, so we consider only sets A whose complexity does not exceed $\mathrm{poly}(n)$; any sufficiently large polynomial can be used here (in fact, $4n$ is enough). So we assume that the sums in (1) and (2), and in similar formulas in the sequel are always restricted to sets $A \subset \mathbb{B}^n$ that have complexity at most $4n$, and take this modified version of (1) as a final definition for $p(y|x)$.

▶ **Definition 4.** Let x be a binary string and let d be an integer. The set of all strings y such that $p(y|x) \geqslant 2^{-d}$ is called *probabilistic prediction d-neighborhood of x.*

We are ready to state the main result of this section.

▶ **Theorem 5.**

(a) *For every n-bit string x and for every d the algorithmic prediction d-neighborhood is contained in probabilistic prediction $d + O(\log n)$-neighborhood.*

(b) *For every n-bit string x and for every d the probabilistic prediction d-neighborhood of x is contained in algorithmic prediction $d + O(\log n)$-neighborhood.*

The next section contains the proof of this result; later we show some its possible extensions.

2.3 The Proof of the Theorem 5

Proof of (a). This direction is simple. Assume that some string y belongs to the algorithmic prediction d-neighborhood of x, i.e., there is a set A containing x and y such that $C(A) + \log|A| \leqslant C(x) + d$. We may assume without loss of generality that $d \leqslant 2n$ otherwise all n-bit string belong to probabilistic prediction d-neighborhood of x (take $A = \mathbb{B}^n$). Then the inequality for $C(A) + \log|A|$ implies that complexity of A does not exceed $4n$, so the set A is included in the sum. This inequality implies also that

$$\frac{\mathbf{m}\,(A)/|A|}{\mathbf{m}\,(x)} \geqslant 2^{-d-O(\log n)}$$

(as we have said, $-\log \mathbf{m}\,(u)$ equals $C(u) + O(\log C(u))$). This fraction is one of terms in the sum that defines $p(y|x)$, so y belongs to the probabilistic prediction $d+O(\log n)$-neighborhood of x. ◀

Before proving the second part (b), we need to prove a technical lemma. It is inspired by [11, Lemma 6] where it was shown that if a string belongs to many sets of bounded complexity, then one of them has even smaller complexity. We generalize that result as follows.

▶ **Lemma 6.** *Assume that sets L and R consist of finite objects (in particular, Kolmogorov complexity $C(v)$ is defined for $v \in L$). Assume that R is has at most 2^n elements. Let G be* *a finite bipartite graph where L and R are the sets of its left and right nodes, respectively. Assume that a right node x has at least 2^k neighbors of Kolmogorov complexity at most i. Then x has a neighbor of complexity at most $i - k + O(C(G) + \log(k + i + n))$. Here $C(G)$ stands for the length of the shortest program that given any $v \in L$ outputs a list of its neighbors.*

Proof. Let us enumerate left nodes that have complexity at most i. We start a selection process: some of them are marked (=selected) immediately after they appear in the enumeration. This selection should satisfy the following requirements:

- at any moment every right node that has at least 2^k neighbors among enumerated nodes, has a marked neighbor;
- the total number of marked nodes does not exceed $2^{i-k}p(i, k, n)$ for some polynomial p (fixed in advance).

If we have such a selection strategy of complexity $C(G) + O(\log(i + k + n))$, this implies that the right node x has a neighbor of complexity at most

$$i - k + O(C(G) + \log(k + i + n)),$$

namely, any its marked neighbor (that marked neighbor can be specified by its number in the list of all marked nodes).

To prove the existence of such a strategy, let us consider the following game. The game is played by two players, who alternate moves. The maximal number of moves is 2^i. At each move the first player plays a left node, and the second player replies saying whether she marks that node or not. The second player loses if the number of marked nodes exceeds $2^{i-k+1}(n + 1)\ln 2$ or if after some of her moves there exists a right node y that has at least 2^k neighbors among the nodes chosen by the first player but has no marked neighbor. (The choice of the bound $2^{i-k+1}(n + 1)\ln 2$ will be clear from the probabilistic estimate below.) Otherwise she wins.

Assume first that the set L of left nodes is finite (recall that the set of right nodes is finite by assumption). Then our game is a finite game with full information, an hence one of the players has a winning strategy. We claim that the second player can win. If it is not the case, the first player has a winning strategy. We get a contradiction by showing that the second player has a probabilistic strategy that wins with positive probability against any strategy of the first player. So we assume that some strategy of the first player is fixed, and consider the following simple probabilistic strategy of the second player: every node presented by the first player is marked with probability $p = 2^{-k}(n + 1)\ln 2$. The expected number of marked nodes is $p2^i = 2^{i-k}(n + 1)\ln 2$. By Markov's inequality, the number of marked nodes exceeds the expectation by a factor of 2 with probability less than $\frac{1}{2}$. So it is enough to show that the second bad case (after some move there exists a right node y that has 2^k neighbors among the nodes chosen by first player but has no marked neighbor) happens with probability at most $\frac{1}{2}$.

For that, it is enough to show that for every node right node y the probability of this bad event is less than $\frac{1}{2}$ divided by the number $|R|$ of right nodes. Let us estimate this probability. If y has 2^k (or more) neighbors, the second player had (at least) 2^k chances to mark its neighbor (when these 2^k nodes were presented by the first player), and the probability to miss all 2^k these chances is at most $(1-p)^{2^k}$. The choice of p guarantees that this probability is less than $2^{-n-1} \leqslant (1/2)/|R|$. Indeed, using the bound $1 - x \leqslant e^{-x}$, it is easy to show that

$$(1 - p)^{2^k} \leqslant e^{\ln 2 \cdot (-n-1)} = 2^{-n-1}.$$

We have proven that the winning strategy exists but have not yet estimated is complexity. A winning strategy can be found be an exhaustive search among all the strategies. The set of all strategies is finite and the game is specified by G, i and k. Therefore the complexity of the first found winning strategy is at most $C(G) + O(\log(i + k))$.

Thus the Lemma 6 is proven in the case when L is a finite set. To extend the proof to general case, notice that the winning condition depends only on the neighborhood of each left node. The worst graph for the the second player is the following "model" graph. It has 2^{2^n+i} left nodes and 2^n right nodes and each of 2^{2^n} possible neighborhoods is shared by 2^i left nodes. A winning strategy for such a graph can be found from n, i and k and hence its complexity is logarithmic in $n + i + k$. That strategy can be translated to the game associated with the initial graph, this translation increases the complexity by $C(G)$, as we have to translate each left node played by the first player to a left node of the model graph. ◀

Having in mind future applications in Subsection 2.4, we will consider in the next statement an arbitrary decidable family $\mathcal{A}$ of finite sets though in this section we need only the case when $\mathcal{A}$ contains all finite sets.

▶ **Corollary 7.** *Let $\mathcal{A}$ be a decidable family of finite sets. Assume that $x_1, \ldots, x_l$ are strings of length n. Denote by $\mathcal{A}^n_m$ all subsets of $\mathbb{B}^n$ of complexity at most m. Then the sum*

$$S := \sum_{A \in \mathcal{A}^n_m,\ x_1,\ldots,x_l \in A} \frac{\mathbf{m}(A)}{|A|}$$

equals to its maximal term up to a factor of $2^{O(\log(n+m+l))}$.

Proof of the corollary. Let M denote the maximal term in the sum S. Obviously the sum S is equal to the sum over $i \leqslant m$ and $j \leqslant n$ of sums

$$\sum_{\substack{A \in \mathcal{A}^n_m \\ C(A)=i \\ \log|A|=j \\ x_1,\ldots,x_l \in A}} \frac{\mathbf{m}(A)}{|A|}. \tag{3}$$

As there are $(m + 1)(n + 1)$ such sums, we only need to prove that each sum (3) is at most $M \cdot 2^{O(\log n+m+l)}$. In other words, we have to show that for all i, j there is a set $H \in \mathcal{A}^n_m$ with $x_1, \ldots, x_l \in A$ such that $\frac{\mathbf{m}(H)}{|H|}$ is greater than the sum (3) up to a factor of $2^{O(\log(n+m+l))}$.

To this end fix i and j. Since $\mathbf{m}(u) = 2^{-C(u)-O(\log C(u))}$, the sum (3) equals

$$\sum_{\substack{A \in \mathcal{A}^n_m \\ C(A)=i \\ \log|A|=j \\ x_1,\ldots,x_l \in A}} 2^{-C(A)-\log|A|+O(\log(n+m))} = \sum_{\substack{A \in \mathcal{A}^n_m \\ C(A)=i \\ \log|A|=j \\ x_1,\ldots,x_l \in A}} 2^{-i-j+O(\log(n+m))} \tag{4}$$

All the terms in the sum (4) coincide and thus the sum (4) is equal to $2^{-i-j+O(\log(n+m))}$ times the number of sets $A \in \mathcal{A}^n_m$ with $C(A) = i$, $\log|A| = j$, $x_1, \ldots, x_l \in A$. Let k denote the floor of the binary logarithm of that number.

Consider the bipartite graph whose left nodes are finite subsets from $\mathcal{A}^n$ of cardinality at most 2^j, right nodes are l-tuples of n-bit strings and a left node A is adjacent to a right node $\langle x_1, \ldots, x_l\rangle$ if all $x_1, \ldots, x_l$ are in A. The complexity of this graph is $O(\log(n + l + j))$ and the logarithm of the number of right nodes is nl. By Lemma 6 there is a set $H \in \mathcal{A}^n_m$ of log-size j and complexity at most $i-k+O(\log(i+j+k+n+l)) = i-k+O(\log(l+m+n))$ with $x_1,\dots,x_l \in A$. The fraction $\frac{\mathbf{m}(H)}{|H|}$ is equal to $2^{-(i-k)-j}$ up to a factor of $2^{O(\log(n+m+l))}$. Recall that the sum (4) equals to $2^k 2^{-i-j}$ up to the same factor and thus we are done. ◀

▶ Remark. Consider the following case of Corollary 7: $\mathcal{A}$ is the family off all finite subsets, $l=1$. As was shown in Subsection 2.2 the sum $\sum_{A\ni x}\mathbf{m}(A)/|A|$, is equal to $\mathbf{m}(x)$ up to a *constant* factor.

By this reason, we expect that the accuracy in the corollary can be improved.

Proof of (b). Let y be some string that belongs to probability prediction d-neighborhood for x. According to (2), it implies that

$$\sum_{A\ni x,y}\frac{m(A)}{|A|} \geqslant \mathbf{m}(x)2^{-d-O(\log n)} = 2^{-d-C(x)-O(\log n)}.$$

Now we will use Corollary 7 for $l=2$, $x_1=x$, $x_2=y$, $m=4n$ and the family of all sets as $\mathcal{A}$. By this corollary there is a set $A\ni x,y$ such that $\mathbf{m}(A)/|A| = 2^{-d-C(x)-O(\log n)}$, so: $C(A)+\log|A|-C(x) \leqslant d+O(\log n)$, i. e. y belongs to the algorithmic prediction $d+O(\log n)$-neighborhood of x. ◀

2.4 Sets of Restricted Type

In some cases we know *a priori* what sets could be possible explanations, and are interested only in models from this class. To take this into account, we consider some family $\mathcal{A}$ of finite sets, and look for sets A in $\mathcal{A}$ that contain the data string x and are “good models” for x. This approach was used in [11]; it turns out that many results of algorithmic statistics can be extended to this case (though sometimes we get weaker versions with more complicated proofs).

In this section we show that Theorem 5 also has an analog for arbitrary decidable family $\mathcal{A}$. The family of all subsets of $\mathbb{B}^n$ that belong to $\mathcal{A}$ is denoted by $\mathcal{A}^n$.

First we consider the case when for each string x the set $\mathcal{A}$ contains the singleton $\{x\}$.

Let us define probability prediction neighborhood for a n-bit string x with respect to $\mathcal{A}$. Again we consider a two-stage process: first, some set of n-bit strings from $\mathcal{A}$ is chosen with probability $\mathbf{m}(A)$. Second, a random element in A is chosen uniformly. Again, we have to assume that we choose sets whose complexity is not greater than $4n$. A value $p_{\mathcal{A}}(y|x)$ is then defined as the conditional probability of $y\in A$ with the condition “the output of the two-stage process is x”:

$$p_{\mathcal{A}}(y|x) = \frac{\sum_{A\ni x,y}\mathbf{m}(A)/|A|}{\sum_{A\ni x}\mathbf{m}(A)/|A|} \tag{5}$$

Here the sum is taken over all sets in $\mathcal{A}^n$ that have complexity at most $4n$.

Again as in Subsection 2.2 the denominator equals $\mathbf{m}(x)$ up to $O(1)$-factor (because $\{x\}\in\mathcal{A}$), so:

$$p_{\mathcal{A}}(y|x) = \frac{\sum_{A\ni x,y}\mathbf{m}(A)/|A|}{\mathbf{m}(x)} \tag{6}$$

up to $O(1)$-factor.

Then $\mathcal{A}$-*probabilistic prediction* d-*neighborhood* is defined naturally: a string y belongs to this neighborhood if $p_{\mathcal{A}}(y|x)\geqslant 2^{-d}$. The $\mathcal{A}$-*algorithmic prediction* d-*neighborhood* for x is defined as follows: a string y belong to it if there is a set $A \ni x, y$ that belongs to $\mathcal{A}^n$ such that $\delta(x, A) \leqslant d$.

Now we are ready to state an analog of Theorem 5:

▶ **Theorem 8.** *Let $\mathcal{A}$ be a decidable family of binary strings containing all singletons. Then:*

(a) *For every n-bit string x and for every d the $\mathcal{A}$-algorithmic prediction d-neighborhood is contained in $\mathcal{A}$-probabilistic prediction $d + O(\log n)$-neighborhood.*

(b) *For every n-bit string x and for every d the $\mathcal{A}$-probabilistic prediction d-neighborhood of x is contained in $\mathcal{A}$-algorithmic prediction $d + O(\log n)$-neighborhood.*

Proof of (a). The proof is similar to the proof of Theorem 5 (a). Assume that a string y belongs to the algorithmic prediction d-neighborhood for x, i.e., there is a set $A \in \mathcal{A}^n$ containing x and y such that $C(A) + \log |A| \leqslant C(x) + d$. If $d > 3n$, then the statement is trivial. Indeed, there is a set $A' \in \mathcal{A}^n$ that contains x and y such that $\delta(x, A') \leqslant 3n$. To prove this, we can not set $A' = \mathbb{B}^n$ any more, as this set may not belong to $\mathcal{A}$. However we may let A' be the first set in $\mathcal{A}^n$, that contains x and y. The complexity of this set is not greater than $|x| + |y| \leqslant 2n$ and log-size is not greater than n. Thus $\delta(x, A') \leqslant 3n$. The rest of the proof is completely similar to the proof of Theorem 5 (a). ◀

Proof of (b). The proof is similar to the proof of Theorem 5 (b). ◀

Now we state and prove Theorem 8 in general case (for families $\mathcal{A}$ that may not contain all singletons). In the case $x \in \bigcup \mathcal{A}^n$, where $n = |x|$, the definition of $\mathcal{A}$-probability prediction neighborhood remains the same. Otherwise, if $x \notin \bigcup \mathcal{A}^n$, the string x can not appear in the two-stage process, so in this case we define $\mathcal{A}$-probability prediction d-neighborhood for x as the empty set for every d. Notice, that now we can not rewrite (5) as (6) because $\{x\}$ may not belongs to $\mathcal{A}$.

Now we define $\mathcal{A}$-algorithmic prediction neighborhood. There is a subtle point that should be taken into account: it may happen that there is no set $A \in \mathcal{A}$ containing x such that $\delta(x, A) \approx 0$. By this reason we include in the algorithmic prediction neighborhood of x the union of all sets A in $\mathcal{A}$, such that $\delta(x, A)$ is as small as it is possible:

▶ **Definition 9.** Let $x \in \mathbb{B}^n$ be a binary string, let d be some integer and let $\mathcal{A}$ be some family of sets. The union of all finite sets in $\mathcal{A}^n$ such that $x \in A$ and every $B \in \mathcal{A}^n$ that contains x satisfies the inequality: $\delta(x, A) \leqslant \delta(x, B) + d$ is called *$\mathcal{A}$-algorithmic prediction d-neighborhood of x*. (In other words, d-neighborhood includes all sets A whose $\delta(x, A)$ is at most d more than the minimum.)

▶ **Theorem 10.** *Let $\mathcal{A}$ be a decidable family of binary strings. Then:*

(a) *For every n-bit string x and for every d the $\mathcal{A}$-algorithmic prediction d-neighborhood is contained in $\mathcal{A}$-probabilistic prediction $d + O(\log n)$-neighborhood.*

(b) *For every n-bit string x and for every d the $\mathcal{A}$-probabilistic prediction d-neighborhood of x is contained in $\mathcal{A}$-algorithmic prediction $d + O(\log n)$-neighborhood.*

Notice that if $x \notin \bigcup \mathcal{A}^n$ then both algorithmic and prediction neighborhoods are empty and the statement is trivial. Therefore in the proof we will assume that this is not the case.

Proof of (a). The proof is completely similar to the proof of Theorem 8. ◀

Proof of (b). Let y be some strings that belongs to probability prediction d-neighborhood for x, that is,

$$\sum_{A \ni x,y} \frac{\mathbf{m}(A)}{|A|} \geqslant 2^{-d} \sum_{A \ni x} \frac{\mathbf{m}(A)}{|A|} \tag{7}$$

Let

$$A_x = \arg\max\{\mathbf{m}(A)/|A| \mid x \in A \in \mathcal{A}^n\}$$

and

$$A_{xy} = \arg\max\{\mathbf{m}(A)/|A| \mid x, y \in A \in \mathcal{A}^n\}.$$

Recall that $\frac{\mathbf{m}(A)}{|A|} = 2^{-C(A)-\log|A|}$ (up to a $2^{O(\log n)}$ factor) and by Corollary 7 the sums in both parts of the equality are equal to their largest terms (again up to $2^{O(\log n)}$ factor). Therefore,

$$2^{-C(A_{x,y})-\log|A_{x,y}|} \geqslant 2^{-d-O(\log n)} 2^{-C(A_x)-\log|A_x|},$$

which means that $\delta(x, A_{x,y}) \leqslant \delta(x, A_x) + d + O(\log n)$. Hence y belongs $\mathcal{A}$-algorithmic prediction $d + O(\log n)$-neighborhood of x. ◀

2.5 Prediction for Several Examples

Consider the following situation: we have not one but several strings $x_1, \ldots, x_l \in \mathbb{B}^n$ that are experimental data. We know that they were drawn independently with respect to the uniform probability distribution in some unknown set A. We want to explain these observation data, i. e. to find an appropriate set A. Again we measure the quality of explanations by two parameters: $C(A)$ and $\log|A|$.

In this section we will extend previous results to this scenario. Again we assume that we know a priori which sets could be possible explanations. So, we consider only sets from a decidable family of sets $\mathcal{A}$.

Let $\overrightarrow{x}$ denote the tuple $x_1, \ldots, x_l$. Let $A \subset \mathbb{B}^n$ be a set that contains all strings from $\overrightarrow{x}$. Then we can restore $\overrightarrow{x}$ from A and indexes of strings from $\overrightarrow{x}$ in A and hence we have :

$$C(\overrightarrow{x}) \leqslant C(A) + l\log|A| + O(\log n).$$

Therefore it is natural to define the *optimality deficiency* of $A \ni \overrightarrow{x}$ by the formula

$$\delta(\overrightarrow{x}, A) := C(A) + l\log|A| - C(\overrightarrow{x}).$$

The definitions of the $\mathcal{A}$-algorithmic prediction d-neighborhood of the tuple $\overrightarrow{x}$ is obtained from Definition 9 by changing x to $\overrightarrow{x}$.

In a similar way we modify the definition of the $\mathcal{A}$-probabilistic prediction neighborhood. Again we consider a two-stage process: first, a set of n-bit strings from $\mathcal{A}$ is chosen with probability $\mathbf{m}(A)$. Second, l random elements in A are chosen uniformly and independently on each other. Again, by technical reason, we assume, that we consider only sets whose complexity is not greater then $(l+3)n$. The value $p_{\mathcal{A}}(y|\overrightarrow{x})$ is defined as the conditional probability of $y \in A$ under the condition [the output of this two-stage process is equal to $\overrightarrow{x}$]:

$$p_{\mathcal{A}}(y|\overrightarrow{x}) = \frac{\sum_{A \ni \overrightarrow{x}, y} \mathbf{m}(A)/|A|^l}{\sum_{A \ni \overrightarrow{x}} \mathbf{m}(A)/|A|^l}.$$

Here both sums are taken over all sets $A \in \mathcal{A}^n$ that have complexity at most $n(l+3)$. (If no such set contains x then $p_{\mathcal{A}}(y|\overrightarrow{x}) = 0$.) By definition, a string y belongs to $\mathcal{A}$-probabilistic prediction d-neighborhood for $\overrightarrow{x}$ if $p_{\mathcal{A}}(y|\overrightarrow{x}) \geqslant 2^{-d}$.

Now we are ready to state an analog of Theorem 10:

▶ **Theorem 11.** *Let $\mathcal{A}$ be a decidable family of binary strings. Then:*

(a) *For every l n-bit strings $\overrightarrow{x}$ and for every d the $\mathcal{A}$-algorithmic prediction d-neighborhood is contained in $\mathcal{A}$-probabilistic prediction $d + O(\log(n + l))$-neighborhood of $\overrightarrow{x}$.*

(b) *For every l n-bit strings $\overrightarrow{x}$ and for every d the $\mathcal{A}$-probabilistic prediction d-neighborhood of $\overrightarrow{x}$ is contained in $\mathcal{A}$-algorithmic prediction $d + O(\log(n + l))$-neighborhood of $\overrightarrow{x}$.*

Proof. The proof is entirely similar to the proof of Theorem 10, but now Corollary 7 is applied for l and $l + 1$ strings so the accuracy becomes $O(\log(n + l))$. ◀

3 Non-uniform Probability Distributions

We have considered so far only uniform probability distributions as statistical hypotheses. The paper [10, Appendix II] justifies such a restriction: it was observed there that for every data string x and for probability distribution P there is a finite set $A \ni x$ that is not worse than P as an explanation for x (with logarithmic accuracy). However, if the data consists of more than one string, then this is not the case. Now, we will explain this in more details.

The quality of a probability distribution P as an explanation for the data $x_1, \dots, x_l$ is measured be the following two parameters:

- the complexity $C(P)$ of the distribution P,
- $-\log(P(x_1) \dots P(x_l))$ (the smaller this parameter is the larger is the likelihood to get the tuple $\overrightarrow{x}$ by independently drawing l strings with respect to P).

We consider only distributions over finite sets such that the probability of every outcome is a rational number. The complexity of such a distribution is defined as the complexity of the set of all pairs $\langle y, P(y)\rangle$ ordered lexicographically.

If P is a uniform distribution over a finite set A then the first parameter becomes $C(A)$ and the second one becomes $-l \log |A|$. If $l = 1$ then for every pair x, P there is a finite set $A \ni x$ such that both $C(A), \log |A|$ are at most $C(P), -\log P(x)$ with the accuracy $O(\log |x|)$. Indeed, let $A = \mathbb{B}^n$ if $P(x) \geqslant 2^{-n}$ and

$$A = \{x \in \mathbb{B}^n \mid P(x) \geqslant 2^{-i}\}$$

if $2^{-i} \leqslant P(x) < 2^{-i+1} \leqslant 2^{-n}$. In both cases we have $C(A) \leqslant C(P) + O(\log n)$ and $\log |A| \leqslant -\log P(x) + 1$.

For $l = 2$ this is not the case:

▶ **Example 12.** Let x_1 be a random string of length $2n$ and $x_2 = 00\dots0y$ be a string of length $2n$ where y is a random string of length n independent of x_1 (that is, $C(x_1, x_2) = 3n+O(1)$). A plausible explanation of such data is the following: the strings x_1, x_2 were drawn independently with the respect the distribution P where half of the probability is uniformly distributed over all strings of length $2n$ and the remaining half is uniformly distributed over all strings of length $2n$ starting with n zeros. The complexity of this distribution P is negligible ($O(\log n)$) and the second parameter $-\log(P(x_1)P(x_2))$ is about $3n$. On the other hand there is no simple set A containing both strings x_1, x_2 with $2 \log |A|$ being close to $3n$. Indeed, for every set A containing x_1 we have $C(A) + \log |A| \geqslant 3n - O(\log n)$ and hence $2 \log |A| \geqslant 6n - 2C(A) - O(\log n) \gg 3n$ (the last inequality holds provided $C(A)$ is small).

Therefore we will not restrict the class of statistical hypotheses to uniform distributions. We will show that the main result of [11] (Theorem 16 below) translates to the case of several strings, i.e., to the case $l > 1$ (Theorem 17 below).

3.1 The Profile of a Tuple $x_1, \ldots, x_l$

Fix $x_1, \ldots, x_l \in \mathbb{B}^n$. As above, we will denote by $\overrightarrow{x}$ the tuple $x_1, \ldots, x_l$. The optimality deficiency is defined by the formula

$$\delta(\overrightarrow{x}, P) = C(P) - \log(P(x_1) \ldots P(x_l)) - C(\overrightarrow{x}).$$

This value is non-negative up to $O(\log(n + l))$, since given P and l we can describe the tuple $\overrightarrow{x}$ in $-\log(P(x_1) \ldots P(x_l)) + O(1)$ bits, using the Shannon-Fano code.

▶ **Definition 13.** The profile $P_{\overrightarrow{x}}$ of the tuple $\overrightarrow{x}$ is defined as the set of all pairs $\langle a, b\rangle$ of naturals such that there is a probability distribution P of Kolmogorov complexity at most a with $\delta(\overrightarrow{x}, P) \leqslant b$.

Loosely speaking, a tuple of strings $\overrightarrow{x}$ is called *stochastic* if there is a simple distribution P such that $\delta(\overrightarrow{x}, P) \approx 0$. In other words, if $(a, b) \in P_{\overrightarrow{x}}$ for $a, b \approx 0$. Otherwise it is called *non-stochastic*. In one-dimensional case non-stochastic objects were studied, for example, in [7], [11]. However, in the one-dimensional case we can not present explicitly a non-stochastic object. In the two-dimensional case the situation is quite different: let x_1 be a random string of length n and let $x_2 = x_1$. For such pair x_1, x_2 there is no simple distribution P with small $\delta(\langle x_1, x_2\rangle, P)$. Indeed, for any probability distribution P we have $C(P) - \log P(x_i) \geqslant C(x_i) = n$ for $i = 1, 2$ (with accuracy $O(\log n)$). Adding these inequalities we get

$$2C(P) - \log(P(x_1)P(x_2)) \geqslant 2n\,.$$

Hence $\delta(\langle x_1, x_2\rangle, P) \geqslant 2n - C(P) - C(x_1, x_2) = n - C(P)$, which is very large provided $C(P) \ll n$.

In general, if strings x_1 and x_2 have much *common information* (i. e. $C(x_1, x_2) \ll C(x_1) + C(x_2)$), then the pair $\langle x_1, x_2\rangle$ is non-stochastic. There is also a non-explicit example of a non-stochastic pair of strings: consider any pair whose first term is non-stochastic. There is no good explanation for the first term, hence there is no good explanation for the whole pair.

The first example suggests the following question: is the profile of the pair of strings x_1, x_2 determined by $C(x_1), C(x_2), C(x_1, x_2)$, P_{x_1}, P_{x_2} and $P_{[x_1,x_2]}$? Here $[x_1, x_2]$ denotes the concatenation of strings x_1 and x_2. Notice that $P_{[x_1,x_2]}$ denotes the 1-dimensional profile of the string $[x_1, x_2]$ and is not to be confused with P_{x_1,x_2}, which is the 2-dimensional profile of the pair of strings x_1, x_2. The following theorem is the main result of Section 3. It provides a negative answer to this question.

▶ **Theorem 14.** *For every n there are strings x_1, x_2, y_1 and y_2 of length $2n$ such that:*

1. *The sets P_{x_1} and P_{y_1}, P_{x_2} and P_{y_2}, $P_{[x_1,x_2]}$ and $P_{[y_1,y_2]}$ are at most $O(\log n)$ apart.*
2. *$C(x_1) = C(y_1) + O(\log n)$, $C(x_2) = C(y_2) + O(\log n)$, $C(x_1, x_2) = C(y_1, y_2) + O(\log n)$.*
3. *However the distance between P_{x_1,x_2} and P_{y_1,y_2} is greater than $0.5n - O(\log n)$. (We say that the distance between two sets R and Q is at most ε if R is contained in ε-neighborhood, with respect to L_∞-norm, of Q, and vice versa.)*

Sketch of proof. Our example is borrowed from [1], where there are several examples of *stochastic* pairs of strings with non-extractable common information.

Consider a finite field $\mathbb{F}$ of cardinality 2^n and a plane (two-dimensional vector space) over $\mathbb{F}$. Let y_1 be a random line on this plane, and y_2 be a random point on this line. Then

$$C(y_1) = 2n, C(y_2) = 2n, C(y_1, y_2) = 3n$$

(everything with logarithmic accuracy). These strings y_1, y_2 have about n bits of common information. On the other hand [1, Theorem 8] states that this common information is non-extractable. By this reason $(1.5n, 0.5n - O(\log n)) \notin P_{y_1,y_2}$.

It is easy to construct another pair of strings x_1, x_2 that has the same properties except that the pair $(n + O(1), O(1))$ is inside P_{x_1,x_2}. To this end let x_1, x_2 be random strings of length $2n$ that share first n bits: $x_1 = x^*x_1^*$, $x_2 = x^*x_2^*$ and $C(x^*x_1^*x_2^*) = 3n + O(1)$. ◀

3.2 Randomness Deficiency

In this subsection we introduce multi-dimensional randomness deficiency and show that the main result of [11] relating 1-dimensional randomness deficiency and optimality deficiency translates to any number of strings.

The 1-dimensional randomness deficiency of a string x in a finite set A was defined by Kolmogorov as $d(x|A) = \log|A| - C(x|A)$. It is always non-negative (with $O(\log|x|)$ accuracy), as we can find x from A and the index of x in A. For most elements x in any set A the randomness deficiency of x in A is negligible. More specifically, the fraction of x in A with randomness deficiency greater than β is less than $2^{-\beta}$. The randomness deficiency measures how non-typical looks x in A.

▶ **Definition 15.** The set of all pairs (a, b) such that there is a set $A \ni x$ of complexity at most a and $d(x|A) \leqslant b$ is called the *stochasticity profile* of x and is denoted by Q_x

To distinguish profiles P_x and Q_x we will call P_x the *optimality profile* in the sequel. Surprisingly, the sets P_x and Q_x almost coincide:

▶ **Theorem 16** ([11]). *For every string x of length n the distance between P_x and Q_x is at most $O(\log n)$.*

The multi-dimensional randomness deficiency is defined in the following way. For a tuple of strings $\overrightarrow{x} = x_1, \dots, x_l$ and a distribution P let

$$d(\overrightarrow{x}|P) = -\log(P(x_1)\dots P(x_l)) - C(x_1, \dots, x_l|P).$$

If $l = 1$ and P is a uniform distribution in a finite set then this definition is equivalently to the one-dimensional case. The randomness deficiency measures how implausible is to get $x_1, \dots, x_l$ as a result of l independent draws from A. The set off all pairs (a, b) such that there is a distribution P of complexity at most a and $d(\overrightarrow{x}|P) \leqslant b$ is called the *l-dimensional stochasticity profile of $\overrightarrow{x}$* and is denoted by $Q_{\overrightarrow{x}}$.

It turns out that Theorem 16 translates to multi-dimensional case:

▶ **Theorem 17.** *For every tuple $\overrightarrow{x} = x_1, \dots, x_l$ of strings of length n the distance between sets $P_{\overrightarrow{x}}$ and $Q_{\overrightarrow{x}}$ is at most $O(\log(n + l))$.*

▶ Remark. Theorem 17 is basically an analog of Theorem 16 for a restricted class of distributions, namely, for product distributions Q on l-tuples, i.e., distributions of the form $Q(x_1, \dots, x_l) = P(x_1) \cdots P(x_l)$. A natural question is whether Theorem 16 can be generalized to any decidable class of distributions. This is indeed the case and the proof is very similar to the proof of Theorem 17.

An Open Question

Can we improve the accuracy in Corollary 7 from $2^{O(\log(n+m+l))}$ to $2^{O(\log(n+l))}$?

Acknowledgments. I would like to thank Alexander Shen and Nikolay Vereshchagin for useful discussions, advice and remarks.

References

1 A. Chernov, An. Muchnik, A. Romashchenko, A. Shen, and N. Vereshchagin. Upper semilattice of binary strings with the relation “x is simple conditional to y”. Theoretical Computer Science 271 (2002) 69–95.

2 P. Gács, J. Tromp, P.M.B. Vitányi. Algorithmic statistics, *IEEE Trans. Inform. Th.*, 47:6 (2001), 2443–2463.

3 A.N. Kolmogorov, Talk at the Information Theory Symposium in Tallinn, Estonia, 1974.

4 Moshe Koppel *Complexity, depth and sophistication* Complex Systems 1, pp. 87–91

5 Li M., Vitányi P., *An Introduction to Kolmogorov complexity and its applications*, 3rd ed., Springer, 2008 (1 ed., 1993; 2 ed., 1997), xxiii+790 pp. ISBN 978-0-387-49820-1.

6 A. Shen *Around Kolmogorov complexity: basic notions and results* `http://arxiv.org/abs/1504.04955`

7 A. Shen *The concept of* (α, β)*-stochasticity in the Kolmogorov sense, and its properties. Soviet Mathematics Doklady, 271(1):295–299, 1983*

8 A. Shen, V. Uspensky, N. Vereshchagin *Kolmogorov complexity and algorithmic randomness.* MCCME, 2013 (Russian). English translation: `http://www.lirmm.fr/~ashen/kolmbook-eng.pdf`

9 N. Vereshchagin, A. Shen *Algorithmic statistics revisited* http://arxiv.org/abs/1504.04950

10 N. Vereshchagin and P. Vitányi *"Kolmogorov's Structure Functions with an Application to the Foundations of Model Selection" . IEEE Transactions on Information Theory 50:12 (2004) 3265-3290. Preliminary version: Proc. 47th IEEE Symp. Found. Comput. Sci., 2002, 751–760.*

11 N.K. Vereshchagin, P.M.B. Vitányi *Rate Distortion a nd Denoising of Individual Data Using Kolmogorov Complexity* IEEE Transactions on Information Theory,56:7 (2010), 3438–3454