On Algorithmic Strong Sufficient Statistics

Nikolay Vereshchagin

Moscow State University, Leninskie gory 1,
Moscow 119991, Russia, ver@mccme.ru
WWW home page: http://lpcs.math.msu.su/~ver

Abstract. The notion of a strong sufficient statistic was introduced in [N. Vereshchagin, Algorithmic Minimal Sufficient Statistic Revisited. *Proc. 5th Conference on Computability in Europe*, CiE 2009, LNCS 5635, pp. 478-487]. In this paper, we give a survey of fine properties of strong sufficient statistics and show that there are strings for which complexity of every strong sufficient statistic is much larger than complexity of its minimal sufficient statistic.

1 Introduction

1.1 Sufficient statistics

Let x be a binary string. A finite set $A \subset \{0,1\}^*$ is called an *(algorithmic) sufficient statistic for* x if $x \in A$ and the sum of the Kolmogorov complexity[1] of A and the binary logarithm of the cardinality of A is close to the Kolmogorov complexity of x:

$$C(A) + \log |A| \approx C(x).$$

More specifically, we call A an *ε-sufficient* statistic for x if the left hand side exceeds the right hand side by at most ε. We do not require the inverse inequality, as it holds with precision $O(\log C(x))$ anyway.

For every x the singleton $\{x\}$ is $O(1)$-sufficient statistic for x. The complexity of this statistic is about $C(x)$. If x is a random string of length n (that is, $C(x) \approx n$) then there is a $O(\log n)$-sufficient statistic for x of much lower complexity: the set of all strings of length n, whose complexity is about $\log n$, is a $O(\log n)$-statistic for x. We will think further of ε as having the order $O(\log n)$ and call such values *negligible*.

1.2 Sufficient statistics and useful information

Sufficient statistics for x are usually thought to capture all the "useful" information from x. The explanation is the following. Let A be a sufficient statistic for x.

[1] Kolmogorov complexity of finite subsets of $\{0,1\}^*$ is defined as follows. We fix any computable bijection $B \mapsto [B]$ from the family of all finite subsets of $\{0,1\}^*$ to the set of binary strings, called a *encoding*. Then we define $C(A)$ as the complexity $C([A])$ of the code $[A]$ of A.

One can show that in this case both the *randomness deficiency* $\log|A| - C(x|A)$ of x in A and $C(A|x)$ are negligible.[2] Let z be the binary notation of the ordinal number of x in A (with respect to the lexicographical order on A). As $C(A|x)$ is negligible, both conditional complexities $C(x|A,z)$ and $C(A,z|x)$ are also negligible.[3] Speaking informally, the two part code (A,z) of x has the same information, as x itself, and its second part z is a string of length $\log|A|$ that is random conditional to its first part A. (Indeed, $C(z|A)$ is up to an additive constant equal to $C(x|A)$, which is close to $\log|A|$.) This encourages us to qualify z as an accidental information (noise) in the pair (A,z), and hence in x. In other words, all useful information from x is captured by the set A.

1.3 Minimal sufficient statistics

If x has a sufficient statistic A of complexity i and log-cardinality j (so that $i+j \approx C(x)$) then for every $k \leqslant j$ it has a sufficient statistic B of complexity $i+k$ and log-cardinality $j-k$ (with logarithmic precision, the complexity of B is actually $i+k+O(\log j)$). This was observed in [3, 2, 5]: the set B is obtained by partitioning A into subsets of size at most 2^{j-k} and considering the part containing x. Thus the most valuable sufficient statistic is the one that has smallest complexity and largest cardinality. Such statistic are informally called *minimal sufficient statistics, MSS,* for x. MSS of x are often considered, as the models extracting all useful information from x and having no noise.

When trying to define the notion of a MSS formally, we face the following problem: for certain strings x for every "negligible" ε a negligible increase of ε may cause a large decrease of the minimal complexity of ε-sufficient statistics for x. For such x is not clear which value of ε to choose in the definition of ε-sufficient statistic and the notion of MSS cannot be defined in a meaningful way. In this paper we will focus on strings for which this is not the case. To define more carefully what it means, consider for a given string x its *structure set* P_x. It consists of all pairs (i,j) of natural numbers for which x has an (i,j)-description, where an *(i,j)-description* is any set $A \ni x$ with $C(A) \leqslant i$ and $\log|A| \leqslant j$. The boundary of P_x is the graph of the function $h_x(i) = \min\{j \mid (i,j) \in P_x\}$, called the *structure function* of x. For every x the boundary of P_x lies above the *sufficiency line* (with logarithmic precision), which by definition consists of all pairs (i,j) with $i+j = C(x)$ (the dash line on Fig. 1). Sufficient statistics correspond to those pairs (i,j) from P_x that are close to the sufficiency line. We will say (quite informally) that *a string x has an MSS*, if there is a natural i with $h_x(i) \approx C(x) - i$ and $h_x(i') \gg C(x) - i'$ for all i' which are "significantly less" than i. Notice that by observation from [3, 2, 5] mentioned above, in this case we also have $h_x(i') \approx C(x) - i'$ for all $i \leqslant i' \leqslant C(x)$ (with logarithmic precision).

---

[2] $C(x|A)$ and $C(A|x)$ are defined as $C(x|[A])$ and $C([A]|x)$, respectively, where $A \mapsto [A]$ is a fixed computable encoding of sets by strings (see the previous footnote).

[3] $C(x|A,z)$ is defined as $C(x|[[A],z])$, where $(x,y) \mapsto [x,y]$ is a computable bijection between pairs of strings and strings; the notation $C(x|A,z)$ is understood in a similar way.

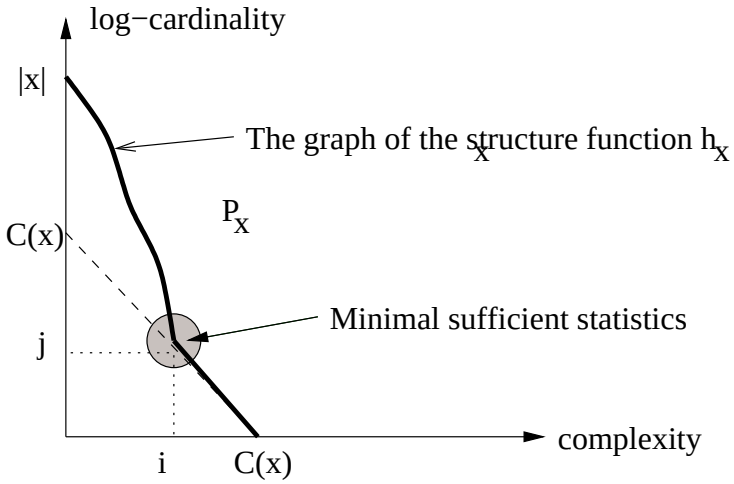

Fig. 1. The structure function h_x. The complexity and log-cardinality of minimal sufficient statistics for x are i and j, respectively.

Example 1. Let y be a string whose structure function h_y leaves the sufficiency line at the point $(C(y), 0)$ (so that $\{y\}$ is essentially the only sufficient statistic for y), see Fig. 2(a).[4] Let $x = [y, z]$, where y is a string of length m that is random conditional to y (that is, $C(z|y) \approx m$). Intuitively, x is obtained from y by adding m bits of noise and y captures all useful information from x. One can show ([7]) that the set P_x looks as drawn on Fig. 2(b).[5] Consider set $A = \{[y, z'] \mid |z'| = m\}$ as a model for x. This model is a $(C(y)+O(\log m), m)$-description of x and hence a MSS for x. The information in A is almost the same as in y, which fact supports the viewpoint that MSS for x capture all useful information from x.

1.4 Universal sufficient statistics

However, as discovered in [2, 6], for every string x that has an MSS there is a MSS that can hardly be considered as a denoised version of x. To define such MSS, fix an algorithm $\mathcal{A}$ that for any given natural k enumerates (in some order) all strings of complexity at most k. Let N_k stand for the number of such strings and let $N_k = 2^{j_1} + 2^{j_2} + \cdots + 2^{j_s}$ be its binary expansion, where $j_1 > j_2 > \cdots > j_s$.

[4] One can show ([6]) that for every decreasing function $h : \{0, 1, \ldots, k\} \to \mathbb{N}$ with $h(0) \leqslant n$ and $h(k) = 0$ there is a string y of length n for which the boundary of the set P_y is at the distance at most $O(\log n)$ from the graph of h.

[5] More specifically, the set P_x is $O(\varepsilon + \log(C(x) + m + j))$-close to the set

$$\{(i, j) \mid (j \leqslant m \Rightarrow i + j \geqslant C(x)) \ \wedge \ (j \geqslant m \Rightarrow (i, j - m) \in P_x)\}.$$

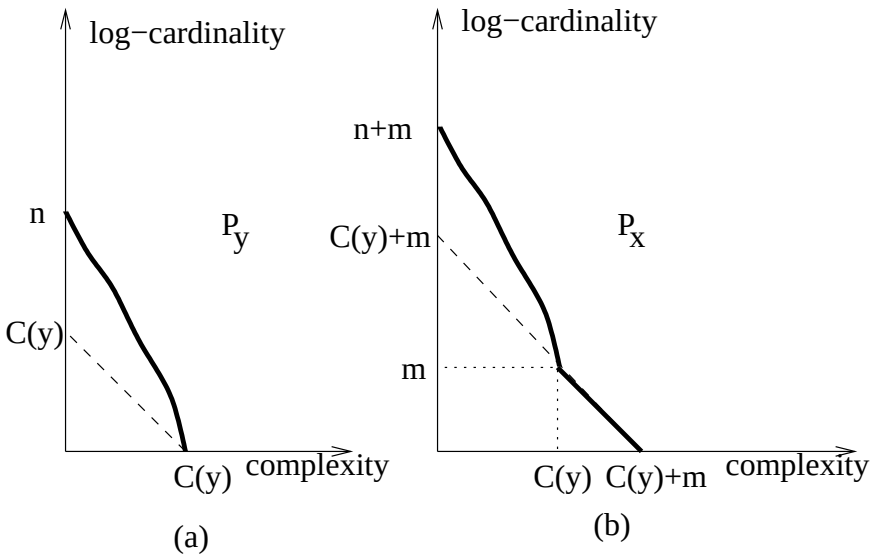

Fig. 2. The sets P_y and P_x

Partition the list of strings enumerated by $\mathcal{A}(k)$ into 2^{j_1} first enumerated strings, 2^{j_2} strings enumerated after them and so on. Let S_{k,j_1}, S_{k,j_2}, …, S_{k,j_s} denote the obtained parts.

By definition $|S_{k,j}| = 2^j$ and it is not hard to show that $C(S_{k,j}) \leqslant k - j + O\log k)$. Assume that $k \geqslant C(x)$ and k is close to $C(x)$. And assume that x belongs to the part $S_{k,j}$. In this case $S_{k,j}$ is a sufficient statistic for x, as $C(S_{k,j}) + \log|S_{k,j}| \leqslant (k - j + O\log k)) + j \approx C(x)$. One can show ([7]) that for every x which has an MSS, for some k close to $C(x)$ and for some j the set $S_{k,j}$ is a MSS for x. This fact is discouraging, because the family $S_{k,j}$ has only two parameters k, j. It implies that for all strings x from Example 1 there are k, j such that the set $S_{k,j}$ is also an MSS for x (where $k \approx C(x) \approx C(y) + m$ and $j \approx m$). Intuitively $S_{k,j}$ has no information about x, and on the other hand one can show that both conditional complexities $C(S_{k,j}|y)$ and $C(y|S_{k,j})$ are negligible. (See [7] for more details.)

1.5 Total conditional complexity

Thus we have to explain why it happens that the model A from Example 1 has the same information, as the model $S_{k,j}$. Also we would like to identify a property of MSS allowing to distinguish the model A from Example 1 from the model $S_{k,j}$.

The first question is easy to answer: we implicitly assumed that u and v have the same information, if both $C(u|v)$ and $C(v|u)$ are negligible. Under this assumption every string x has the same information, as its shortest description x^*.

In the context of separating the information into a useful one and an accidental one, such an assumption is certainly wrong. Indeed, the entire information in x^* (being a random string) is a noise, while x may have useful information. In algorithmic statistics, it is more helpful to think that u and v have the same information only if *total* conditional complexities $CT(u|v)$ are $CT(v|u)$ are negligible. The total conditional complexity $CT(u|v)$ is defined as the minimal length of a total program p for u conditional to v: $CT(u|v) = \min\{|p| \mid U(p,v) = u$ and $U(p,z)$ halts for all $z\}$ (here U is the universal Turing machine). The total conditional complexity can be much larger than the ordinary one.[6] If both $CT(u|v)$ are $CT(v|u)$ are negligible, then their structure sets P_u and P_v are close to each other and they have similar algorithmic-statistical properties. We will call such strings *equivalent* in the sequel.

1.6 Strong sufficient statistics and their fine properties

To answer the second question, the paper [7] introduced a notion of a *strong* sufficient statistics. We call $A \ni x$ a *strong* statistic (or model) for x if $CT(A|x)$ is negligible. As we mentioned, the sufficiency requirement implies that ordinary (not total) conditional complexity $C(A|x)$ is negligible. For strong sufficient statistics we additionally require the total conditional complexity $CT(A|x)$ be negligible. More specifically, we call A *ε-strong* model for x if $CT(A|x) \leqslant \varepsilon$ and we call A an *ε-good* model for x if A is ε-strong and ε-sufficient for x.

It easy to see that A is a strong model for x iff both total complexities $CT(x|A,z)$, $CT(A,z|x)$ are negligible, where z is the ordinal number of x in A. Indeed, given the pair (A,z) we can find x by means of a short total program (even, if A is not strong). Conversely, if A is a strong statistic for x, than from x we can compute A by means of a short total program and then compute the ordinal number of x in A.

Strong sufficient statistics have the following nice properties.

(a) The model A from Example 1 is a good MSS for x. Indeed, given x we can find A by a constant length total program that maps $[y,z]$ to the set $\{[y,z'] \mid |z'| = |z|\}$. That is, x has a good MSS if and only if x is equivalent to a string of the form specified in Example 1.

(b) Strong MSS are unique: if both A, B are strong MSS for x, then $CT(A|B) \approx CT(B|A) \approx 0$ (Theorem 6 in [7]). In particular, if B is any strong MSS for the string x from Example 1 and A is the model of x from that example, then $CT(A|B) \approx CT(B|A) \approx 0$. (We state here the result in a highly informal way, for the precise statement see [7].)

(c) Good statistics satisfy the observation from [3, 2, 5]: If x has a good sufficient statistic A of complexity i and log-cardinality j, then for every $k \leqslant j$ it has

[6] In particular, it is not hard to show that for all n there is string x of length n with $CT(x|p) \geqslant n/3 - O(1)$ for every shortest description p of x. Moreover, this inequality holds for every description p for x of length at most $C(x) + n/3$. On the other hand, by a result of [1], for every x of length n there is a description p for x with $CT(p|x) = O(\log n)$ and $|p| \leqslant C(x) + O(1)$.

a good statistic B of complexity $i+k$ and log-cardinality $j-k$ (with logarithmic precision): again, the set B is obtained by partitioning A into subsets of size at most 2^{j-k} and considering the part containing x. Thus the most valuable good statistic is that one that has smallest complexity and largest cardinality.

1.7 Our result

Recall that one of the goal of introducing the notion of a good MSS is to separate MSS from Example 1 from MSS of the form $S_{k,j}$. We conjecture that this is true: there are stings x that have a MSS but have no ε-strong MSS of the form $S_{k,j}$ for some $\varepsilon = \Omega(|x|)$. In this paper we answer another question left open in [7]: is it true that every string that has a MSS has also a good MSS? We show that this is not the case: there are strings that have MSS but all their strong sufficient statistics have much larger complexity than that of MSS.

2 Preliminaries

We denote by $\{0,1\}^*$ the set of all strings over the binary alphabet $\{0,1\}$, and by $|x|$ the length of a string x.

The Kolmogorov complexity $C(x)$ of a binary string x and conditional Kolmogorov complexity $C(x|y)$ of a binary string x given another string y are defined as follows. We call a Turing machine U with input alphabet $0,1$ *universal* if for every other Turing machine V with the same input alphabet there is a binary string c_V such that $U(c_V p, y) = V(p, y)$ for all binary strings p, y. The notation $V(p, y)$ refers to the output of machine V when run on input binary strings p, y separated by the blank symbol. Fix a universal machine U. The conditional Kolmogorov complexity is defined as

$$C(x|y) = \min\{|p| \mid p \in \{0,1\}^*,\ U(p, y) = x\}.$$

If $U(p, y) = x$, we say that p is a program (or description) of x conditional to y. The value $C(x|y)$ depends not only of x but also on U. However, if we change U to another universal machine V then $C(x|y)$ is changed by at most an additive constant. Unconditional Kolmogorov complexity $C(x)$ is defined as $C(x|\text{empty string})$.

We will use in the sequel without reference the following properties of Kolmogorov complexity:

- The number of strings of Kolmogorov complexity less than k is less than 2^k.
- $C(x) \leqslant |x| + c$, $C(x|y) \leqslant C(x) + c$, for some c and all x, y;
- For every computable function f mapping strings to strings there is c such that $C(f(x)|x) \leqslant c$ and $C(f(x)) \leqslant C(x) + c$ for all x;
- (Conditional version of the previous inequality.) For every computable function f mapping pairs of strings to strings there is c such that $C(f(x, y)|y) \leqslant C(x|y) + c$ for all x, y;

– Symmetry of information: $C(x,y) \approx C(x)+C(y|x)$. This equality holds with "logarithmic precision". Specifically, we have

$$C(x,y) \leqslant C(x) + C(y|x) + 2\log\min\{C(x), C(y|x)\} + c$$

for some c and all x, y, and

$$C(x) + C(y|x) \leqslant C(x,y) + 4\log(C(x) + C(y|x)) + c.$$

– (Conditional version of symmetry of information). For all x, y, z,

$$C(x,y|z) \approx C(x|z) + C(y|x,z).$$

Here one inequality is true up to a $2\log\min\{C(x|z), C(y|x,z)\}+c$ error term and the other one up to a $4\log(C(x|z) + C(y|x,z)) + c$ error term.

3 Results

Our results establish existence of strings x that have MSS but all their strong sufficient statistics have much larger complexity than that of MSS.

Theorem 1. *Assume that integer numbers i, j, l satisfy the inequalities*

$$l \leqslant i, \quad i + j \leqslant n - 3.$$

Then there is a string x of length n and complexity $i + j + O(\log n)$ that

(a) has a $(i + O(\log n), j)$-description,
(b) has no $(i, n - i - 3)$-descriptions, and
(c) has no l-strong $(i + j, n - i - j - 3)$-descriptions.

Fig. 3 visualises this theorem. Fig. 3(a) shows the set P_x: item (a) of Theorem 1 is responsible for the right sloping segment of the boundary of P_x and item (b) of Theorem 1 is responsible for the left sloping segment of the boundary of P_x. Fig. 3(b) shows the similar set P_x^l consisting by definition of all pairs (i, j) such that x has an l-strong i, j-description:

$$P_x^l = \{(i,j) \mid (\exists A \ni x)\ C(A) \leqslant i,\ \log|A| \leqslant j,\ CT(A|x) \leqslant l\}.$$

The complexity of every l-strong sufficient statistic of x is at least (about) j bits more than that of MSS, which is about i. Notice that this is the best separation possible, as the singleton $\{x\}$ is a $O(1)$-strong sufficient statistic of complexity about $C(x)$ for every x, and in our case $C(x)$ is about j bits more than i.

Let in Theorem 1 $l = i = j = n/3$, say. Then the string x existing by the theorem has an MSS of complexity $n/3$ while each all $n/3$-strong $n/3$-sufficient statistics have complexity at least $2n/3$.

Theorem 1 does not claim anything about how rare are such strings x. Such strings a rare, as for majority of strings x of length n the set $\{0,1\}^n$ is a good MSS

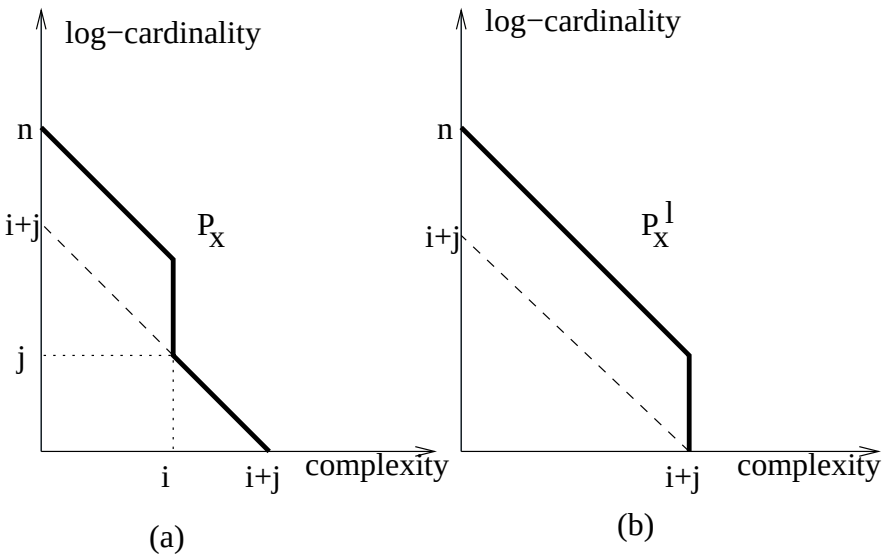

Fig. 3. The sets P_x and P_x^l

for x. A more meaningful question is whether such strings might appear with high probability in a statistical experiment. More specifically, assume that we sample a string x in a given set $A \subset \{0,1\}^n$, where all elements are equiprobable. Might it happen that with high probability (say with probability 99%) A is a MSS for x and at the same time x has no strong MSS? The following theorem provides an affirmative answer to this question.

Theorem 2. *Assume that integer i, j, l, k satisfy the inequalities*

$$l \leqslant i, \quad i + j \leqslant n - 3, \quad k \leqslant j.$$

Then there is set $A \subset \{0,1\}^n$ of cardinality between $2^j/n$ and 2^j and complexity at most $i + O(k + \log n)$ such that all but 2^{j-k} its elements x have complexity $i + j + O(k + \log n)$, have no $i, (n - i - 3)$-descriptions, and have no l-strong $(i + j), (n - i - j - 3)$-descriptions.

If $k = 2 \log n$, say, then the set A is a MSS for a majority of its elements. Indeed, the structure set of all but $|A|/n$ elements from A has the shape shown on Fig. 3(a). On the other hand, the set P_x^l has the shape shown on Fig. 3(b) for those elements.

Remark 1. Regarding Theorem 2 one can wonder whether there are "bad" sets A such that for a majority of $x \in A$, A is not a sufficient statistic for x. An example of such A is the set $\{0,1\}^n \setminus \{y\}$ where y is a random string of length n (that is, $C(y) \approx n$). The complexity of A is close to n and thus $C(A) + \log |A| \approx 2n$.

Hence A is not a sufficient statistics for all its elements. Notice that for all $x \in A$ the set $\{0,1\}^n$ is a strong MSS for x.

References

1. Bruno Bauwens, Anton Makhlin, Nikolay Vereshchagin, Marius Zimand. Short lists with short programs in short time. ECCC report TR13-007. http://eccc.hpi-web.de/report/2013/007/
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. M. Li and P.M.B. Vitányi, *An Introduction to Kolmogorov Complexity and its Applications* , Springer-Verlag, New York, 2nd Edition, 1997.
5. A.Kh. Shen, Discussion on Kolmogorov complexity and statistical analysis, *The Computer Journal*, 42:4(1999), 340–342.
6. N.K. Vereshchagin and P.M.B. Vitányi, Kolmogorov's structure functions and model selection, *IEEE Trans. Information Theory*, 50:12 (2004) 3265-3290.
7. N. Vereshchagin, Algorithmic Minimal Sufficient Statistic Revisited In: *5th Conference on Computability in Europe*, CiE 2009, Heidelberg, Germany, July 19–24, 2009. Proceedings. LNCS 5635. pp. p. 478-487.

A Appendix: Proofs

We start with the following observation.

Lemma 1. *Assume that A is a l-strong statistic for a string x of length n. Let $y = [A]$ be the code of A. Then y has a $(l + O(1), n)$-description.*

Proof. Let p be a string of length less than l such that $U(p, x)$ is defined for all strings x of length n. Consider the set $\{U(p, x) \mid x \in \{0,1\}^n\}$. Its cardinality is at most 2^n and complexity at most $l + O(1)$. If $CT(y|x) \leqslant l$ for some $x \in \{0,1\}^n$ then there is p such that y belongs to such a set and hence y has a $(l + O(1), n)$-description.

We prove now Theorem 1. Consider the family $\mathcal{B}$ consisting of all sets $B \subset \{0,1\}^*$ with $C(B) \leqslant i + j$, $\log |B| \leqslant n - i - j - 3$ and the family $\mathcal{D}$ consisting of all sets sets D with $C(D) \leqslant i$, $\log |D| \leqslant n - i - 3$. By Lemma 1 it suffices to find a string x of length n and a set $A \subset \{0,1\}^n$ with $x \in A$ such that

(a) $C(A) \leqslant i + O(\log n)$, $\log |A| \leqslant j$,
(b) there is no $B \in \mathcal{B}$ which includes x and whose code $[B]$ has a $(l + O(1)), n$-description,
(c) there is no set $D \in \mathcal{D}$ that includes x.

The complexity of x will be $i + j + O(\log n)$ automatically. Indeed, $C(x)$ is at most that much, as A is its $(i + O(\log n), j)$-description. On the other hand, if

$C(x)$ were less than $i+j-O(1)$, then the singleton $\{x\}$ would be its $O(1)$-strong $(i+j),0$-description, which contradicts to (b).

The set A will be chosen "in several attempts". We start with A being the set of 2^j first strings of length n. Then we start the enumeration of sets in families $\mathcal{B},\mathcal{C},\mathcal{D}$ (by running the universal machine U in parallel on all inputs). The crucial fact is that sets from $\mathcal{C}$ appear in this enumeration in at most $2^{l+O(1)}$ portions (each portion has at most 2^n sets). On any step s of this process we keep the following information:

- the family $\mathcal{C}_s$ of all sets in $\mathcal{C}$ discovered so far (up to step s),
- the family $\mathcal{B}_s$ of all sets in $\mathcal{B}$ discovered so far.
- the family $\mathcal{D}_s$ of all sets in $\mathcal{D}$ discovered so far.

On each step s, after having updated $\mathcal{B}_s,\mathcal{C}_s,\mathcal{D}_s$ we update the set A (if needed) and denote the resulting set by A_s. We will always have $|A_s|\leqslant 2^j$. We update A on a step s, if the set A becomes covered by the union of sets from the family $(\mathcal{C}_s\cap\mathcal{B}_s)\cup\mathcal{D}_s$. This can only happen if on step s either a new set in families $\mathcal{B},\mathcal{D}$ or a new portion of sets in $\mathcal{C}$ is discovered. The way we change A (defined later) will ensure that the total number of changes of A is $O(n^2 2^i)$. On some (unknown) step s all sets in $\mathcal{B},\mathcal{C},\mathcal{D}$ will be discovered. The version A_s of A constructed on that step s will be the sought set. Indeed, it has appropriate cardinality, it is not covered by the union of sets in $(\mathcal{C}\cap\mathcal{B})\cup\mathcal{D}$ and may be identified by the number of its changes plus a logarithmic amount of information (numbers i,j,l,n) needed to run the above enumeration. Thus its complexity is at most $i+O(\log n)$.

The algorithm to change A on step s (such that the set A_{s-1} is covered by sets in $(\mathcal{C}_s\cap\mathcal{B}_s)\cup\mathcal{D}_s$) is the following. We choose any set $A\subset\{0,1\}^n$ such that

- $2^j/n\leqslant|A|\leqslant 2^j$,
- every $B\in\mathcal{C}_s$ of cardinality at most 2^{n-j-3} has at most $O(n)$ common strings with A,
- A is disjoint with all sets in $\mathcal{B}_s\cup\mathcal{D}_s$.

The next lemma shows that such A exists.

Lemma 2. *Assume that a family $\mathcal{C}$ consists of at most 2^{2n+c} sets, each of them consisting of at most 2^{n-3-j} elements of a universe U of size 2^{n-1}. Then there is a set $A\subset U$ of cardinality at most 2^j and at least $2^j/n$ that has at most $2n+c+1$ common elements with each $B\in\mathcal{C}$.*

Proof. The set A is chosen at random: each of its 2^j elements $a_1,\ldots,a_{2^j}$ is chosen with uniform distribution in the universe. Elements $a_1,\ldots,a_{2^j}$ are chosen independently, thus some of them may coincide, in which case the cardinality of A is less than 2^j.

We have to show that the statement of the theorem holds with positive probability. To this end note that for every fixed B in $\mathcal{C}$ and for every fixed set of indexes $\{i_1,\ldots,i_{2n+c+1}\}\subset\{1,2,\ldots,2^j\}$ the probability that *all* $a_{i_1},\ldots,a_{i_{2n+c+1}}$ fall in B is at most $(2^{n-j-3}/2^{n-1})^{2n+c+1}=2^{-(j+2)(2n+c+1)}$. The number of sets of indexes as above is at most $(2^j)^{2n+c+1}$ and the cardinality of $\mathcal{C}$ is at most

2^{2n+c}. By union bound the probability that a random set A does not satisfy the lemma is at most

$$2^{2n+c}2^{j(2n+c+1)}2^{-(j+2)(2n+c+1)} < 1/2.$$

Thus the probability that a random set A does not satisfy the second requirement is less than $1/2$.

The first requirement ($|A| > 2^j/n$) does not hold with probability

$$(2^{n-1})^{2^j/n}(2^j/n2^{n-1})^{2^j} < 1/2.$$

(The first factor is an upper bound for the number of $2^j/n$-element subsets of the universe and the second number is an upper bound for the probability that all 2^j randomly chosen strings fall into a fixed set of size $2^j/n$.)

As the universe we consider the set of all strings of length n minus the union of sets in $\mathcal{B}_s \cup \mathcal{D}_s$. Notice that the cardinality of the union of these sets does not exceed

$$2^{i+j+1}2^{n-i-j-3} + 2^{i+1}2^{n-i-3} = 2^{n-1}.$$

(Here $2^{i+j+1}, 2^{i+1}$ are upper bounds for the number of sets in $\mathcal{B}, \mathcal{D}$, respectively, and $2^{n-i-j-2}, 2^{n-i-3}$ are upper bounds for the size of each of them.) Hence we can find an appropriate set A by the lemma.

It remains to estimate the number of A's changes. To this end consider the sequence $s_1 < s_2 < \cdots < s_N$ of all steps on which A has been changed. The number of steps s_m when a new portion of sets in $\mathcal{C}$ is discovered does not exceed $2^{l+O(1)} = O(2^i)$ (the last inequality holds by the assumption). The number of steps s_m when a new set in $\mathcal{D}$ is discovered also does not exceed $O(2^i)$. Thus it suffices to show that the number of steps s_m such that on all steps s between s_m and s_{m+1} neither of those two events happened is $O(n^2 2^i)$.

Consider any such step s_m. Recall that any set from $\mathcal{B}$ has cardinality at most $2^{n-i-j-3} \leqslant 2^{n-j-3}$. By construction the set A_{s_m} has at most $O(n)$ common elements with any set in the family $\mathcal{C}_{s_m} \cap \mathcal{B}$ and is disjoint with any set in $\mathcal{D}_{s_m} \cup \mathcal{B}_{s_m}$. The sets $\mathcal{C}, \mathcal{D}$ have not been changed between steps s_m and s_{m+1}. Therefore by step s_{m+1} all elements of A were covered by sets from $\mathcal{C}_{s_m} \cap (\mathcal{B}_{s_{m+1}} \setminus \mathcal{B}_{s_m})$. Hence the family $\mathcal{B}_{s_{m+1}} \setminus \mathcal{B}_{s_m}$ must have at least $\frac{2^j/n}{O(n)} = \Omega(2^j/n^2)$ sets (recall that $|A_{s_m}| \geqslant 2^j/n$). The total number of sets in $\mathcal{B}$ is at most 2^{i+j+1}, as the complexity of each of them is at most $i+j$. So the total number of A's changes is at most

$$O(2^i) + \frac{2^{i+j+1}}{\Omega(2^j/n^2)} = O(n^2 2^i).$$

Theorem 1 is proved.

Theorem 2 is proved similarly to Theorem 1. The only difference that we change A each time when at least 2^{j-k} strings in A are covered by sets in $(\mathcal{C}_s \cap \mathcal{B}_s) \cup \mathcal{D}_s$. As the result, the number of changes of A will increase 2^k times and the complexity of A will increase by k.