УДК 519.9

КОЛМОГОРОВСКАЯ СЛОЖНОСТЬ КЛАССОВ ОБЩЕРЕКУРСИВНЫХ ФУНКЦИЙ С ОГРАНИЧЕННОЙ ЕМКОСТЬЮ

В.И. Донской

Таврический национальный университет им. В.И. Вернадского
факультет математики и информатики
кафедра информатики
пр-т Вернадского, 4, г. Симферополь, Украина

Abstract.

The double inequality $VCD(S) \leqslant K_l(S) < VCD(S) \log l$ is proven, where $VCD(S)$ is the *Vapnik — Chervonenkis Dimension of some collection of general recursive functions,* $K_l(S)$ *is the Kolmogorov's complexity of this collection* S, l *is the length of data set (sample) which is shattered by* S.

The novel $pVCD$ *approach for* VCD *estimation based on above inequality is suggested.*

Введение

В 1964 г. А.Н. Колмогоров ввёл понятие сложности конечного объекта (например, слова в некотором алфавите) [4]. Он определил *сложность как минимальное число двоичных знаков, содержащих всю информацию о задаваемом объекте*, достаточную для восстановления этого объекта. Такое определение сложности связывалось с развитием алгоритмического подхода к теории случайности и закономерности. Под *закономерностью* А.Н. Колмогоров подразумевал любое алгоритмически проверяемое свойство объектов, присущее лишь узкому их классу (достаточно малому по мере).

Вполне естественно полагать неслучайными те объекты, в которых обнаруживается достаточно много закономерностей. На базе такого подхода можно идентифицировать «*несжимаемость*» объекта со случайностью. Тогда объекты с малой колмогоровской сложностью (малой длиной описания) следует считать неслучайными, а несжимаемые - сложность которых близка к исходной «длине» этих объектов — случайными. Указанный подход можно считать обоснованным, если убедиться в том, что несжимаемые последовательности обладают признаками случайности (стохастичности), известными из теории вероятностей. Такое обоснование получено шведским математиком Мартином — Лёфом [3].

Решение задач обучения распознаванию, извлечения закономерностей, синтеза моделей принятия решений по неполным начальным данным при внимательном рассмотрении оказывается так или иначе направленным на *выбор* достаточно *простых алгоритмов*, позволяющих классифицировать (идентифицировать) объекты. В статистической теории обучения степень такой «простоты» оценивается *сложностью* *семейства алгоритмов*, из которого осуществляется выбор алгоритма, «объясняющего» заданную для обучения выборку.

Сложность классов решающих функций (не обязательно алгоритмических отображений), введенная В.Н. Вапником и А.Я. Червоненкисом [1], — ёмкость или VC — размерность (VCD), — на сегодняшний день является общепризнанной характеристикой классов моделей, используемых при решении задач распознавания и поиска закономерностей [2, 5, 6, 7].

Обсуждение деталей статистической теории обучения, основанной на обеспечении гарантированной равномерной сходимости частот ошибок решающих правил к соответствующим вероятностям по выбранному классу решающих функций, и спорные вопросы остаются за рамками этой статьи. Используется только строго доказанное положение, состоящее в том, что чем меньше VCD (сложность) используемого класса решающих функций (моделей), тем лучше оценка надёжности принятия решения (распознавания, вывода) на основе решающей функции, выбранной из указанного класса [1].

Развитием колмогоровского подхода к поиску закономерностей в данных является так называемый принцип MDL — *Minimum Description Length* (минимума длины описания) [7], состоящий в следующем. Пусть даны данные D и семейство рекурсивно перечислимых моделей. Согласно принципу MDL, выбирается модель с кратчайшим эффективным описанием суммы длины в битах описания модели и длины в битах описания данных, которые могут быть сжаты (закодированы) при помощи этой модели. В работе П. Витаньи и М. Ли [7] показано, что принцип выбора гипотез на основе MDL даёт тот же результат, что и оптимальное байесовское правило, максимизирующее апостериорную условную вероятность.

В данной статье изучается *связь между VCD и колмогоровской сложностью классов рекурсивных функций*, из которых выбираются решения, что является основной *целью* работы. Несмотря на изначальную сугубо теоретическую постановку этой задачи, получен и практически полезный результат — приём оценивания VCD на основе определения длин программ, описывающих закономерности.

Далее используются следующие обозначения:

$\{0,1\}^*$ — множество строк из нулей и единиц любой длины; при необходимости можно считать их упорядоченными (по длине и лексикографически) и отождествлять с натуральными числами $\mathbb{N}_0 = \{0,1,2,\dots\}$ [3];

P_r — класс общерекурсивных функций;

$P_{p.r}$ — класс частично рекурсивных функций;

x, y, p — строки из $\{0,1\}^*$;

отношение между функциями $F(x) \preceq G(x)$ означает, что $F(x) \leqslant G(x) + C$ для некоторой константы C.

1. Колмогоровская сложность конечных объектов и семейств рекурсивных функций

Сложностью строки (текста) по А.Н. Колмогорову можно считать длину самого короткого двоичного слова, содержащего всю информацию, необходимую для восстановления этого текста при помощи какого-нибудь фиксированного способа (алгоритма) декодирования [4].

Определение 1. [3], [4] Условная сложность слова x при известном слове y по произвольной частично рекурсивной функции F есть

$$K_F(x|y) = \begin{cases} \min l(p) : F(p, y) = x; \\ \infty, \text{если } \forall p \in \{0,1\}^* F(p, y) \neq x \end{cases}$$

Слово p такое, что $F(p, y) = x$, называется *кодом или программой*, по которой функция F восстанавливает слово x.

Различные конечные объекты можно сравнивать между собой по колмогоровской сложности, учитывая, что существует частично рекурсивная функция F_0, называемая оптимальной, такая, что для любой частично рекурсивной функции G $K_{F_0}(x|y) \preceq K_G(x|y)$ [3, 4].

Условной сложностью $K(x|y)$ слова x при известном слове y называется сложность $K_{F_0}(x|y)$ по некоторой раз и навсегда фиксированной оптимальной частично рекурсивной функции F_0.

Рассмотрим произвольную общерекурсивную функцию f; если y $= f(x)$, то x и y — числа из расширенного натурального ряда, которые могут быть представлены двоичными строками.

Определение 2.

1. Сложность функции f по аргументу x и по частично рекурсивной функции U есть

$$K_U(f|x) = \begin{cases} \min l(p) : U(p, x) = y; \\ \infty, \text{если } \forall p \in \{0,1\}^* U(p, x) \neq y, \end{cases}$$

где $y = f(x)$.

2. Сложность функции f в области D по частично рекурсивной функции U есть

$$K_{U,D}(f) = \max_{x \in D} K_U(f|x)$$

3. Сложность класса функций $S \subset P_r$ в области D по частично рекурсивной функции U есть

$$K_{U,D}(S) = \max_{f \in S} K_{U,D}(f)$$

4. Сложность класса функций $S \subset P_r$ на области D есть

$$K_D(S) = \min_{U \in P^*_{p.r}} K_{U,D}(S),$$

где $P^*_{p.r}$ — семейство всех частично рекурсивных функций U таких, для которых имеют смысл сложности $K_U(f|x)$, $K_{U,D}(f)$, $K_{U,D}(S)$.

Согласно определению 2, сложность некоторого семейства общерекурсивных функций S в области D есть наименьшая длина двоичного слова, позволяющего вычислить «самую сложную» функцию $f \in S$ при помощи «наилучшей» функции $U \in P^*_{p.r}$ (с кратчайшим кодом p).

В задачах обучения распознаванию свойства P по эмпирической выборке $\tilde{X}_l = X_1, X_2, \ldots, X_l$, состоящей из l векторов размерности n, с указанными значениями $P(X_1), P(X_2), \ldots, P(X_L)$, отыскивается решающая функция (правило) A, позволяющая для произвольного допустимого вектора $X \in D$, $X = x_1, \ldots, x_n$, определить значение предиката (свойства) $P(X)$. Если $A(X) = P(X)$ для всех $X \in \tilde{X}_l$, то правило F называют *корректным по выборке* X_l. Если $A(X) = P(X)$ для всех $X \in D$, то правило A называется *абсолютно точным.*

В рамках настоящей работы искомая функция A полагается частично рекурсивной (алгоритмом - в силу тезиса Чёрча). Такое предположение необходимо в условиях компьютерной реализации распознающих алгоритмов.

Решающая функция A обычно отыскивается в некотором классе S алгоритмов, который определяется применяемым подходом к обучению (линейные модели, решающие деревья, нейронные сети и др.). С позиций теоретической и прикладной информатики, выборку $\tilde{X}_l$ можно, не теряя общности, считать упорядоченным набором $n \times l$ чисел из $\mathbb{N}_0$.

Обозначим X^l — конечное множество любых допустимых обучающих выборок.

Определение 3. 1. Сложность алгоритма A относительно выборки $\tilde{X}_l$ по частично рекурсивной функции U есть

$$K_U(A|\tilde{X}_l) = \begin{cases} \min l(p) : U(p, \tilde{X}_l) = \tilde{y}; \\ \infty, \text{если } \forall p \in \{0,1\}^* U(p, \tilde{X}_l) \neq \tilde{y}, \end{cases}$$

где $\tilde{y} = A(X_1), A(X_2), \ldots, A(X_l)$ - двоичное слово длины l, представляющее некоторое число $\tilde{y}$.

2. Сложность алгоритмического семейства $S \subset P_{p.r}$ на области X^l по частично рекурсивной функции U есть

$$K_{U,l}(S) = \max_{A \in S} \max_{\tilde{X}_l \in X^l} K_U(A|\tilde{X}_l)$$

3. Сложность алгоритмического семейства $S \subset P_{p.r}$ на области X^l есть

$$K_l(S) = \min_{U \in P^*_{p.r}} K_{U,l}(S)$$

2. Сложность классов решающих функций по В.Н. Вапнику

Класс решающих функций S, применяемых к произвольной выборке $\tilde{X}_l = X_1, X_2, \ldots, X_l$, может быть охарактеризован ёмкостью или, как её называют, VCD — размерностью (Vapnik — Chervonenkis Dimension). Емкость, как характеристика сложности класса S, вводится следующим образом [1]. Произвольное решающее правило $A \in S$ определяет двоичный вектор

$$A(X_1), A(X_2), \ldots, A(X_l) \tag{1}$$

— способ разделения выборки на два класса: $C_0 = \{X \in \tilde{X}_l : A(X) = 0\}$ и $C_1 = \{X \in \tilde{X}_l : A(X) = 1\}$. Число различных двоичных векторов — способов разделения выборки $\tilde{X}_l$ всеми правилами $A \in S$ — обозначается $\Delta^S(X_1, \ldots, X_l)$. *Функцией роста* системы S называется

$$m^S(l) = \max_{\tilde{X}_l \in X^l} \Delta^S(X_1, \ldots, X_l),$$

где максимум берётся по всем возможным выборкам $\tilde{X}_l$ длины l.

Доказано [1], что функция роста либо тождественно равна 2^l, либо удовлетворяет неравенству

$$m^S(l) < 1,5\frac{l^{h_S}}{h_S!}$$

при $l > h_S$. Число h_S, характеризующее сложность класса S, называют *ёмкостью* или VCD этого класса.

Емкость класса определяет наибольшую длину выборки $l = h_s$, которую можно разбить на два класса всеми 2^{h_S} способами, взятую по всем возможным выборкам $\tilde{X}_l$ при любом l. Если $h_S \equiv 2^l$, то говорят, что класс S имеет бесконечную ёмкость.

Значение понятия VCD классов решающих правил для статистической теории обучения объясняется неравенством [1]

$$P\left(\sup_{A\in S}|P(A) - \nu(A)| > \varepsilon\right) < 9\frac{(2l)^{h_S}}{h_S!}e^{\frac{-\varepsilon^2 l}{4}},$$

где A — произвольное решающее правило, выбранное из класса S в процессе обучения, $\nu(A)$ — частота ошибок правила A на случайно извлеченной эмпирической выборке $\tilde{X}_l$, $P(A)$ — неизвестная вероятность ошибки правила A на генеральной совокупности объектов X; $\varepsilon > 0$. Конечность VCD h_S обеспечивает равномерную сходимость частот $\nu(A)$ ошибок к вероятностям $P(A)$ при $l \to \infty$;

при $h_S < \infty$:

$$\forall \ \varepsilon > 0 \quad \lim_{l\to\infty} P\left(\sup_{A\in S}|P(A) - \nu(A)| > \varepsilon\right) \to 0$$

Чем меньше ёмкость h_S класса S, тем «быстрее» равномерная сходимость по классу S.

С точки зрения теории равномерной сходимости, для обучения предпочтительны классы решающих правил, имеющих небольшую ёмкость.

3. Связь между сложностью по Колмогорову и VCD классов общерекурсивных функций

Теорема 1. *Пусть система общерекурсивных функций S имеет ограниченную ёмкость h_S и сложность $K_l(S)$. Тогда при $h_S \geqslant 2$*

$$h_S \leqslant K_l(S) < h_S \log l \tag{2}$$

Доказательство. Сложность семейства функций $K_{U,l}(S)$ по произвольной частично рекурсивной функции U определяется с использованием соотношения $U(p, \tilde{X}_l) = \tilde{y}$, в котором значение $\tilde{y}$ есть вариант разбиения выборки $\tilde{X}_l$ на два подмножества. При этом $\tilde{y} = A(\tilde{X}_l)$ для некоторого зафиксированного элемента $A \in S$, и разные варианты разбиения могут быть получены только путём применения к $\tilde{X}_l$ различных функций семейства S. Чтобы выполнялось равенство $U(p, \tilde{X}_l) = A(\tilde{X}_l)$ для всех $A \in S$, аргумент p должен принимать не менее $m^S(l)$ значений, поскольку U является функцией. Из этого следует, что $l(p) \geqslant]\log m^S(l)[$.

Покажем, что

$$\min_{U \in P^*_{p.r}} K_{U,l}(S) =]\log m^S(l)[\tag{3}$$

Для этого, с учетом последнего неравенства, достаточно указать такую функцию $U^* \in P^*_{p.r}$, что $K_{U^*,l}(S) =]\log m^S(l)[$. Функция U^* определяется таблицей 1, содержащей $m^S(l) \times M$ клеток, где M - мощность множества X_l.

Код (номер) программы p	Код (номер) выборки $\tilde{X}_l$				
	$\tilde{X}_l^{(0)}$	$\dots$	$\tilde{X}_l^{(j)}$	$\dots$	$\tilde{X}_l^{(M-1)}$
0	$\dots$	$\dots$	$\dots$	$\dots$	$\dots$
$\dots$	$\dots$	$\dots$	$\dots$	$\dots$	$\dots$
i	$\tilde{y}_{i,0}$	$\dots$	$\tilde{y}_{i,j}$	$\dots$	$\tilde{y}_{i,M-1}$
$\dots$	$\dots$	$\dots$	$\dots$	$\dots$	$\dots$
$m^S(l)$	$\dots$	$\dots$	$\dots$	$\dots$	$\dots$

Таблица 1.

Значения $\tilde{y}_{ij}$, $0 \leqslant i \leqslant m^S(l) - 1$, $0 \leqslant j \leqslant M - 1$, содержащиеся в таблице 1 и являющиеся двоичными кодами длины l (1), отождествляются с соответствующими им числами расширенного натурального ряда; также числами (номерами) интерпретируются выборки $\tilde{X}_l$ и коды p.

Для класса событий S ограниченной ёмкости $h_S \geqslant 2$ и $l > 2$ справедливы соотношения

$$2^{h_S} \leqslant m^S(l) < 1,5\frac{l^{h_S}}{h_S!} < l^{h_S} = 2^{h_S \log l}$$

$$h_S \leqslant \log m^S(l) < h_S \log l$$

С учётом равенства (2), получаем

$$h_S \leqslant K_l(S) < h_S \log l$$

Следствие 1.

$$\frac{K_l(S)}{\log l} < h_S \leqslant K_l(S)$$

Поскольку для любой частично рекурсивной функции $U \in P^*_{p.r}$ выполняется неравенство $K_l(S) \leqslant K_{U,l}(S)$, то $h_S \leqslant K_{U,l}(S)$, и в качестве оценки VCD сверху может быть использовано любое обобщенное описание класса функций (алгоритмов) семейства S - программа p.

Пример 1. Пусть $S^{\mu,n}_{BT}$ — семейство бинарных решающих деревьев не более чем с μ листьями и n переменными. Любое дерево $A \in S^{\mu}_{BT}$ содержит не более $\mu - 1$ внутренних вершин, соответствующих некоторым из n переменных, из которых выходят по два указателя на одну из $\mu - 2$ вершин или номер класса.

номер переменной	левая ссылка (указатель)	правая ссылка (указатель)

Рис. 1. Элемент списка с двумя ссылками

Поэтому дерево A задаётся списком из $(\mu - 1)$ элементов длины не более $\log n + 2\log\mu = \log(\mu^2 n)$, откуда получаем: $l(p) = (\mu - 1)\log(\mu^2 n)$ и

$$h_{S^{\mu,n}_{BT}} < (\mu - 1)\log(\mu^2 n).$$

Сравнивая с результатом, полученным другим способом в [2],

$$h_{S^{\mu,n}_{BT}} < \mu\log(n\mu),$$

обнаруживаем, что оценка VCD класса $S^{\mu,n}_{BT}$ получилась завышенной. Следовательно, построенная программа p не является минимальной. Действительно, при построении линейного описания дерева A можно обойтись списком из элементов только с одним указателем в каждой вершине ветвления, полагая, что вместо второго указателя используется непосредственная конкатенация следующего элемента списка. Тогда $l(p) = (\mu - 1)\log(n\mu)$ и

$$h_{S^{\mu,n}_{BT}} < (\mu - 1)\log(n\mu) \tag{4}$$

Замечательно, что при построении оценки (3) в рассуждении используются приемы программирования: на начальную вершину дерева ссылка не требуется, а две терминальные вершины (0 и 1) могут быть непосредственными операндами и рассматриваться как константы. Поэтому к $\mu - 2$ необходимым адресам добавляются *первыми* эти два адреса.

$$\boxed{\text{номер переменной} \mid \text{ссылка (указатель)}} \rightarrow \text{следующий элемент}$$
$$\downarrow$$

Рис. 2. Элемент списка с одним указателем

4. Программы и выполняющие их алгоритмы

Неравенство (2) определяет мажоранту для VCD h_S системы общерекурсивных функций S. Согласно определению 3 справедливо неравенство

$$h_S \leqslant K_l(S) \leqslant K_{U,l}(S), \tag{5}$$

где U — произвольная частично рекурсивная функция такая, что $U(p, \tilde{X}_l) = \tilde{y}$; $\tilde{y} = A(X_1), A(X_2), \ldots, A(X_l)$, причем в соответствии с (5) выполняется неравенство

$$h_S \leqslant l(p) \tag{6}$$

В силу тезиса Чёрча, U — некоторый алгоритм (машина), который для аргумента $\tilde{X}_l$, по слову (программе) p вычисляет правильный ответ $\tilde{y}$. Следовательно, возможны интерпретации объектов p и U, которые представлены в таблице 2.

p	U
программа на алгоритмическом языке	машина, выполняющая эту программу
таблица команд машины Тьюринга (программа)	машина Тьюринга
код на языке C ++	компьютер
описание в виде двоичной строки p, по которому можно восстановить алгоритм $A \in S$	любой алгоритм (машина), воспроизводящая по описанию p алгоритм A

Таблица 2.

Так в примере 1 программа p есть сжатый код бинарного решающего дерева с μ листьями, использующего булевы переменные из множества $\{x_1, x_2, \ldots, x_n\}$. Интерпретация этого кода p состоит в «проходе» по ссылкам, что, очевидно, реализуемо любым компьютером.

Покажем, что случай $h_S = l(p)$ возможен при такой «программной» оценке VCD системы S.

Пример 2. Пусть S - множество $Mon(n)$ булевых одночленов (конъюнкций, мономиалов) над n переменными, содержащими любое число $k \in \{0, 1, \dots, n\}$ литералов вида $M_n^k = x_{i_1}^{\sigma_1} \& \dots \& x_{i_k}^{\sigma_k}$. Очевидно, $|Mon(n)| = 3^n$; $VCD(Mon(n)) < n \log 3$. В работе [5] доказано, что $VCD(Mon(n)) = n$ и, более того, что $VCD(Mon^+(n)) = VCD(Mon^-(n)) = n$, где $\mathrm{Mo}n^+(n)$ — множество конъюнкций, содержащих только положительные литералы, а $\mathrm{Mo}n^-(n)$ — содержащих только отрицательные литералы. Тогда расшифровка любого мономиала $\mathrm{A} \in \mathrm{Mo}n^+(n)$ обеспечивается по строке $p = p_1, \dots, p_n$, содержащей n бит, соответствующих переменным $x_1, \dots, x_n$, в которой значение $p_i = 1$, если положительный литерал x_i входит в конъюнкцию A, и $p_i = 0$ — если этот литерал не входит в A. Тогда $l(p) = n = h_{Mon^+(n)}$.

Подход к оцениванию на основе соотношения $h_S \leqslant l(p)$: $U(p, \tilde{X}_l) = \tilde{y} = A(\tilde{X}_l)$ будем называть *принципом программирования оценки* VCD, сокращенно — $pVCD$.

Этапы реализации $pVCD$ при оценивании VCD h_S семейства S состоят в следующем.

1. Изучение семейства S и определение минимальной совокупности свойств (параметров, структурных особенностей), конкретное указание которых позволяет для некоторого алгоритма U сформировать двоичное слово p_A такое, что для любого входа X выполняется $U(p_A, X) = A(X)$
2. Определение длины $l(p_A)$ слова p_A как искомой оценки.

Согласно такому определению $pVCD$, любые свойства семейства S, не изменяющиеся при переходе от одного алгоритма из семейства S к другому, не должны учитываться в слове p.

Пример 3. Семейство линейных предикатов (однослойных персептронов) $L(n)$, состоящее из вычислимых предикатов $<< a_1x_1 + a_2x_2 + \dots + a_nx_n \geqslant a_0 >>$, как известно, имеет $VCD(L(n)) = n + 1$. Любой предикат из $L(n)$ полностью определяется словом $p = a_1, a_2, \dots, a_n a_0$ длины $O(n + 1)$. Если $|a_i| \leqslant M \in \mathbb{N}$, $i = 0, 1, \dots, n$, то $l(p) \leqslant (n + 1)(\log M + 1)$.

Подкласс $L_k(n) \subset n$, содержащий предикаты, зависящие не более чем от k из n переменных, при помощи $pVCD$ может быть оценен неравенством

$$VCD(L_k(n)) < k(\log n + \log M + 1) + \log M + 1 = O(k \log n)$$

Пример 4. Класс функций, вычислимых многослойными нейронными сетями с бинарной активацией и m весами имеет VCD, оцениваемую как $O(m \log m)$ [6]. Используя принцип $pVCD$, учитываем m параметров—весов и структуру сети—связи между параметрами, описываемые ссылками (указателями) от элемента сети, содержащего параметр i, к элементу сети, содержащему параметр j. Пусть $\xi_{ij} = 1$, если такая ссылка существует, иначе $\xi_{ij} = 0$. Обозначим

$$C = \max_{i \in \{1,2,\ldots,n\}} \sum_{j=1}^{m} \xi_{ij},$$

полагая константу $C < m$. Тогда

$$l(p) = O(m) + Cm \log m = O(m \log m)$$

Заключение

Основной теоретический результат, полученный в данной работе, состоит в том, что для произвольного класса S общерекурсивных функций $VCD(S)$ и колмогоровская сложность класса S являются величинами одного порядка. Поэтому поиск корректных описаний p_D эмпирических данных D, имеющих малую длину $l(p_D)$, гарантирует ограничение вапниковской сложности класса решающих правил, используемого при расшифровке (распознавании) с использованием описаний p_D. Обратно, выбор решающих правил из семейств, характеризуемых невысокой VC-размерностью, гарантирует ограничение длины описания данных.

Указанный результат обосновывает выбор кратчайших (по длине описания) эмпирических закономерностей в задачах анализа и распознавания данных.

В практическом плане, оценивание VCD величиной сложности по Колмогорову позволило предложить $pVCD$ принцип получения оценки VCD сверху как новую технику, основанную на структурно—программистском подходе.

Представляется перспективным, используя $pVCD$, получить оценки VCD для сложных классов семейств рекурсивных функций; обобщить $pVCD$ на классы субрекурсивных функций.

Большое теоретическое значение имеет дальнейшее изучение связи классов $TIME$- и $SPACE$-сложности частично рекурсивных функций (алгоритмов) U, используемых в определении функции $K_{U,l}(S)$, с величиной h_S VCD класса S.

Список литературы

1. Вапник В.Н. Восстановление зависимостей по эмпирическим данным. — М.: Наука, 1979. — 448 с.
2. Дюличева Ю.Ю. Оценка VCD r—редуцированного эмпирического леса // Таврический вестник информатики и математики. — 2003. — № 1 — С. 31 - 42.
3. Звонкин А.К., Левин Л.А. Сложность конечных объектов и обоснование понятий информации и случайности с помощью теории алгоритмов // Успехи математических наук. — 1970. — Т. XXV, Вып.6. — С. 85 - 127.
4. Колмогоров А.Н. Теория информации и теория алгоритмов. — М.: Наука, 1987. — 304 с.
5. Natschläger T., Schmitt M. Exact VC-Dimension of Boolean Monomials // Information Processing Letters. — 1996. — № 59. — P. 19 - 20.
6. Sontag E.D. VCDimension of Neural Networks // Neural Networks and Machine Learnong. Springer-Verlag, Berlin — 1998. — P. 69 - 95.

7. Vitanyi P.M.B., Li M. Minimum Description Length Induction, Bayesianism, and Kolmogorov Complexity // IEEE Trans. on Inf. Theory. — 2000. — Vol. 46, №2. — P. 446 - 464.