Original Paper *Forma*, **15**, 339–345, 2000

A Good Sampling Method for Guessing Rectangles in $[0, 1]^2$

Masanobu OHTSUKI, Atsushi MINATO and Satoru OZAWA

Division of Applied Synergetics, Graduate School of Science and Engineering, Ibaraki University, 4-12-1 Nakanarusawa, Hitachi 316-8511, Japan

(Received March 28, 2000; Accepted June 12, 2000)

Keywords: Rectangle, Learning, Guessing

Abstract. We introduce a good sampling method for guessing the shape of axis parallel rectangles in $[0, 1]^2$. We will show that with the algorithm based on that good sampling we can guess the rectangles within worst error $O(1/m^{2/3})$, where m is the number of sample points.

1. Introduction

We consider the following mathematical problem.

"How can we guess efficiently the shape of an unknown axis parallel rectangle in $[0, 1]^2$ by sampling ?"

The "sampling" means here to draw some points in $[0, 1]^2$ and examine whether each point is in the unknown rectangle or not. (We suppose that there is a teacher that tells us the information of "in the rectangle" or "out of the rectangle".)

In this paper we introduce a good sampling method for guessing axis parallel rectangles in $[0, 1]^2$ efficiently, the worst error of which is $O(1/m^{2/3})$.

This problem is introduced naturally by the "learning theory".

In the present learning theory, the problem above is interpreted as the problem of finding a "1-stage" efficient learning algorithm with "membership query" only. See VALIANT, 1984; LAIRD, 1988; NATARAJAN, 1991 for the present learning theory.

Here we see the background of the present learning theory.

In that theory a "concept" is thought to be expressed by a figure (precisely speaking, a subset) of some domain and "learning" is identified with "guessing the figure".

And the model is usually introduced at first. That is, learning domain X and the set of the concepts to be learned $\mathscr{C} \subset 2^X$ are defined and queries are defined that is allowed in the learning, such as "membership query" "equivalence query" "subset query" etc.

Then the learnability is defined and it is researched whether $\mathscr{C}$ is learnable in that sense or not.

A "stage" of an algorithm consists of drawing sample points and doing a computation after this sampling, and the number of stages is the number of times a sequence of samples are drawn without any other computation (see LINIAL *et al.*, 1991 for the stage of an algorithm).

Hence a 1-stage algorithm is the one that runs in so called "static sampling mode", where all the necessary sample points are drawn before any computation is performed.

A general algorithm is more likely to be of multistage type in construction where the subsequent sampling and computation may exploit the result of the sampling and the computing in the proceeding stages.

It's clear that a multi-stage algorithm is more powerful than a 1-stage one. But analysis of the error of a multi-stage algorithm is more complicated.

In this paper we deal with only 1-stage algorithms.

The learning of rectangles is studied in BLUMER *et al.* (1989) and NATARAJAN (1991) in PAC (Probably Approximately Correct)-learning sense or studied in multistage algorithms model (see BLUMER *et al.*, 1989 and VALIANT, 1989 for PAC-learning, and see OHTSUKI *et al.*, 1999 for a good 2-stage algorithm and its application).

But it is not studied precisely in the model defined bellow, in which the guessing is done with a 1-stage algorithm, without any stochastic phenomenon, and with sampling only, and the error is measured as a function of m, the number of sample points.

Now we give the guessing model more precisely.

Let $\mathscr{P}$ be the set of all the axis parallel rectangles in $[0, 1]^2$.

That is $\mathscr{P} \stackrel{def}{=} \{[a_1, a_2] \times [b_1, b_2];\ 0 \le a_1 \le a_2 \le 1,\ 0 \le b_1 \le b_2 \le 1\}$.

And let $\mathscr{A}$ be an algorithm for guessing $c \in \mathscr{P}$.

Guessing a figure of $c \in \mathscr{P}$ by $\mathscr{A}$ is done as follows.

At first $c \in \mathscr{P}$ is chosen. The algorithm $\mathscr{A}$ does not know what c is, but knows that it is surely a figure in $\mathscr{P}$.

To guess the figure, $\mathscr{A}$ can ask the teacher machine $\mathscr{T}$ with pointing any point in $[0, 1]^2$ whether it is in c or not. Such a point is called a "sample point" or simply a "sample".

For each question, $\mathscr{T}$ gives the guessing algorithm the correct information "1" or "0", where "1" means that the point is in the rectangle, and "0" means it is out of the rectangle.

After researching m sample points, $\mathscr{A}$ puts out h as a guessed result (h is called a "hypothesis" in the learning theory).

Of course h does not always correspond to c, and the error between h and c, denoted by $error_{\mathscr{A}}(c)$, is measured by the area of $(h \triangle c)$, where $\triangle$ is the symmetric difference, i.e. $h \triangle c = (h - c) \cup (c - h)$.

We call such an algorithm $\mathscr{A}$ a "guessing(learning) algorithm for $\mathscr{P}$" or say "$\mathscr{A}$ guesses(learns) $\mathscr{P}$".

Now we define the error of an algorithm $\mathscr{A}$, and the inevitable error of the set $\mathscr{P}$ as follows.

Definition 1

$e_{\mathscr{A}}(m) \stackrel{def}{=} \sup\,\{error_{\mathscr{A}}(c) = area(c \triangle \mathscr{A}(c, m));\ c \in \mathscr{P}\}$, *where* $\mathscr{A}(c, m)$ *is the guessed result of the algorithm* $\mathscr{A}$ *when* $\mathscr{A}$ *guesses* c *with* m *sample points.*

$e_{\mathscr{P}}(m) \stackrel{def}{=} \inf\,\{e_{\mathscr{A}}(m);\ \mathscr{A}$ *is an algorithm for guessing* $\mathscr{P}\}$

□ (*Definition 1*)

In the definition above, $e_{\mathscr{A}}(m)$ is the error of an algorithm $\mathscr{A}$, and $e_{\mathscr{P}}(m)$ is the error that is inevitable when we guess c in $\mathscr{P}$ with any 1-stage algorithm, i.e. it is the error caused by the difficulty of guessing the shapes of figures in $\mathscr{P}$.

That is, we can measure the difficulty of $\mathscr{P}$ by $e_{\mathscr{P}}(m)$,and it is clear that $e_{\mathscr{P}}(m) \le e_{\mathscr{A}}(m)$ for any algorithm $\mathscr{A}$.

It is very difficult to know precisely about $e_{\mathscr{P}}(m)$ and it is not the main subject in this paper.

The main subject in the present paper is to construct an efficient algorithm for guessing $\mathscr{P}$, the error of which comes near to $e_{\mathscr{P}}(m)$.

Now we see an example of sampling for guessing an unknown rectangle in $[0, 1]^2$.

An example is shown in Figs. 1, 2 and 3. The rectangle to be guessed is, for example, c in Fig. 1 and a guessing algorithm draws up to m sample points like Fig. 2, i.e. it draws sample points regularly along each axis with equal width. And finally it puts out a guessed result h in Fig. 3.

In these figures the points "×"'s are the points that are in the rectangle c in Fig. 1, and "· "'s are the points out of c.

Let us call this simple guessing algorithm $\mathscr{A}_0$.

It can be easily shown that $e_{\mathscr{A}_0}(m) = O(1/m^{1/2})$ that is $e_{\mathscr{A}_0}(m) \le k/m^{1/2}$ for some constant k.

And that means $e_{\mathscr{P}}(m) = O(1/m^{1/2})$. (See GRAHAM *et al.*, 1989 for the notation "$O()$".)

We will construct a more efficient algorithm in the next section.

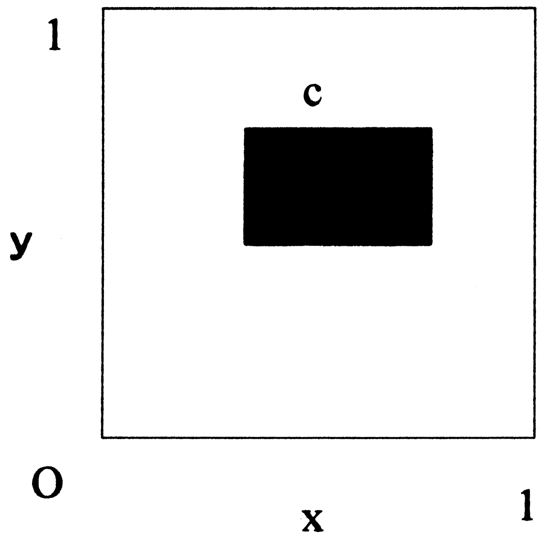

Fig. 1. An example of a rectangle to be guessed.

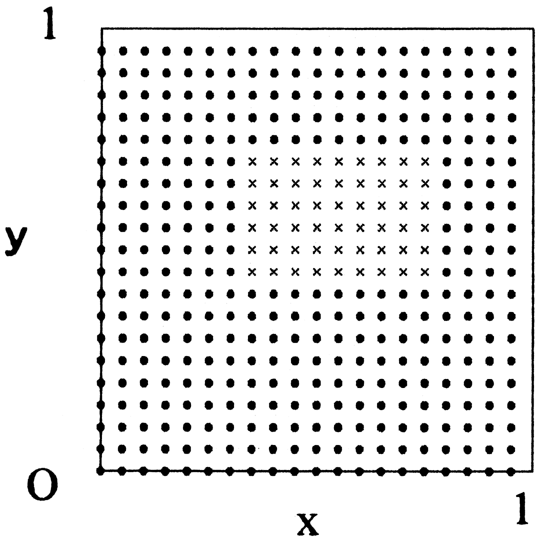

Fig. 2. An example of sample points (m = 400).

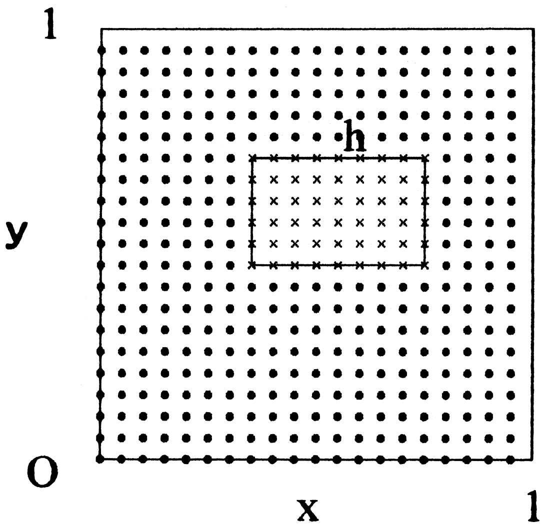

Fig. 3. An example of a guessed result.

2. A Good Sampling Method for Guessing Rectangles in $[0, 1]^2$

In this section we introduce a good sampling method for guessing rectangles in $[0, 1]^2$, construct an algorithm based on it, and show that the error of this algorithm is $O(1/m^{2/3})$.

That means that $e_{\mathscr{P}}(m)$ is $O(1/m^{2/3})$.

We show the following theorem, the proof of which includes a good sampling method and the efficient algorithm based on that sampling.

Theorem 1 $e_{\mathscr{P}}(m) = O(1/m^{2/3})$

<Proof>

We will prove $e_{\mathscr{P}}(m) \le \alpha/m^{2/3}$ (for some constant α), by constructing a good algorithm for guessing $\mathscr{P}$.

It is an algorithm which draws all the points in the set *Spoints*(m) (defined below) as sample points and puts out the minimum rectangle as a guessed one that includes all the sample points that turned out to be in the unknown rectangle. We call this algorithm $\mathscr{A}_1$.

Let $q = q(m)$ be a positive integer, $M = M(m)$ be a positive real number, and $d = d(m) = 1/q(m)$.

For a positive integer m, define a set of sample points $Spoints_x(m)$, $Spoints_y(m)$ as follows.

$Spoints_x(m) \overset{def}{=} \{(x, y);\ x = id, y = j/(i+1)M\}, 0 \le i \le q, 1 \le j \le (i+1) M, i, j: integer\}$

$Spoints_y(m) \overset{def}{=} \{(x, y);\ x = j/(i+1)M, y = id,\ 0 \le i \le q,\ 1 \le j \le (i+1)M, i, j: integer\}$

And define $Spoints(m) \overset{def}{=} Spoints_x(m) \cup Spoints_y(m)$

If $m \ge 64$, we can take $q(m) = \lfloor m^{1/3} \rfloor$, $M = (1/2)\, m^{1/3}$ and $|\ Spoints(m)\ | \le 2Mq^2 \le m$.

Some examples of this set *Spoints*(m) are shown in Fig. 4.

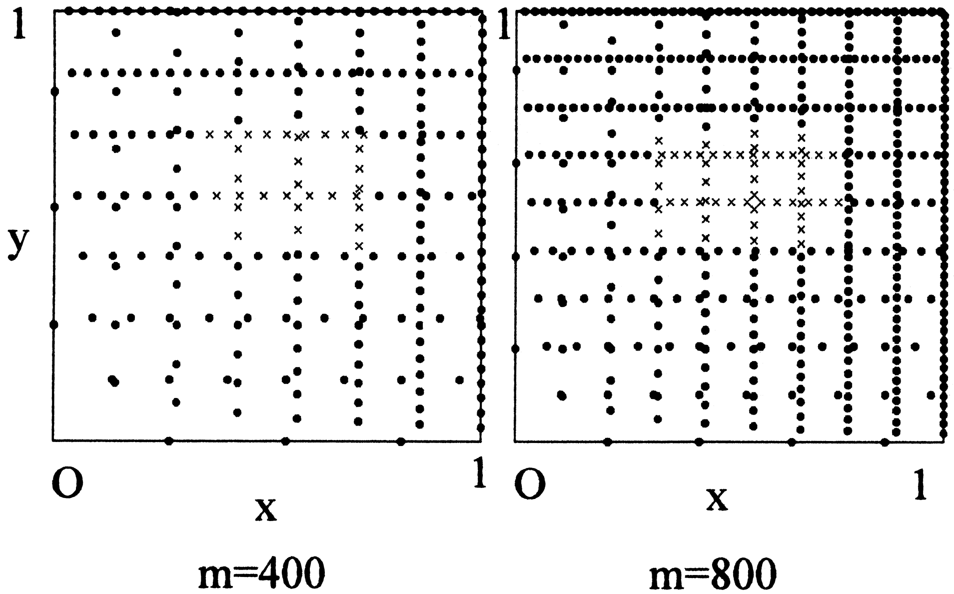

Fig. 4. *Spoints*(m)(m = 400, m = 800).

Now we show the following fact (*) at first.

(*)$\forall$ *rectangle* $c \subset [0, 1]^2$, $c \cap Spoints(m) = \phi \Rightarrow area(c) \leq \alpha/m^{2/3}$, for some constant α, and $area(c)$ is the area of the rectangle c.

Let S be $S \overset{def}{=} c \cap Spoints(m)$. We prove that $S = \phi \Rightarrow area(c) \leq \alpha/m^{2/3}$.

Let $c = [a_1, a_2] \times [b_1, b_2]$ be a rectangle $\subset [0, 1]^2$ such that $c \cap Spoints(m) = \phi$.

There exists a non-negative integer i such that $id = i/q \leq (b_2 - b_1) < (i + 1)d = (i + 1)/q$.

Now two cases should be considered.

(Case 1)[$i = 0$]

In this case $(b_2 - b_1) < 1/q \leq 1/M$. (We assume $m \geq 8$.) Then $\exists j \geq 1$; $1/\{(j + 1) M\} \leq (b_2 - b_1) < 1/(jM)$. Since $(a_2 - a_1) > jd \Rightarrow S \neq \phi$, so $(a_2 - a_1) \leq jd$, hence $(a_2 - a_1)(b_2 - b_1) \leq d/M$.

(Case2) [$i \geq 1$]

In this case $(a_2 - a_1) < 1/\{(i + 1)M\}$, thus $(a_2 - a_1)(b_2 - b_{\_1}) < d/M$.

By the research of all the cases and the definition of $M(m)$, $\bar{d}(m)$, it has been shown that $S = \phi \Rightarrow area(c) \ \leq = \alpha/m^{2/3}$ for some constant α.

We note here the following fact.

We call h a "consistent (hypothesis) with c on the sample points $\{z_1, z_2, ..., z_m\}$" if $I_h(z_i) = I_c(z_i)$ for $i = 1, 2, ..., m$, where $I_u()$ is a characteristic function on some domain u, i.e. $I_u(z) = 1$ if $z \in u$, and $I_u(z) = 0$ if $z \notin u$.

For any rectangle $c \subset [0, 1]^2$, and $h \subset [0, 1]^2$ that is consistent with c on $S = \{z_1, z_2, ..., z_m\}$, $c \triangle h$ can be expressed by the union of up to 4 rectangles that do not include any points in S.

Then it is clear that (*) above implies that $\forall c \subset [0, 1]^2$, $area(c \triangle \mathscr{A}_1(c, m)) \leq 4\alpha/m^{2/3}$ because $\mathscr{A}_1$ is an algorithm which puts out a consistent rectangle with c on $Spoints(m)$. That means $e_{\mathscr{A}1}(m) = O(1/m^{2/3})$, and hence $e_{\mathscr{P}}(m) = O(1/m^{2/3})$. $\square$

3. Conclusion

We have introduced a good way of sampling for guessing axis parallel rectangles in $[0, 1]^2$ to show that $e_{\mathscr{P}}(m) = O(1/m^{2/3})$.

Some problems are left to the future research.

1. Extend the way of sampling introduced in this paper to the n-dimensional case. That is, construct an efficient algorithm for guessing n-dimensional rectangles in $[0, 1]^n$.
2. Estimate $e_{\mathscr{P}}(m)$ more sharply. That is, construct a more efficient algorithm than the algorithm introduced in this paper.

REFERENCES

BLUMER, A., EHRENFEUCHT, A., HAUSSLER, D., MANFRED, K. and WARMUTH, K. (1989) Learnability and the Vapnik-Chervonenkis dimension, *Journal of the ACM*, **36**, No. 4, pp. 929–965.

GRAHAM, R. L., KNUTH, D. E. and PATASHNIK, O. (1989) *Concrete Mathematics*, Addison-Wesley Publishing Company, pp. 407–418.

LAIRD, P. D. (1988) *Learning from Good and Bad Data*, Kluwer Academic Publishers, Boston, pp. 1–213.

LINIAL, L., MANSOUR, Y. and RIVEST, R. L. (1991) Results on learnability and the Vapnik-Chervonenkis

dimension, *Information and Computation 90*, Academic Press, pp. 33–49.
NATARAJAN, B. (1991) *Machine Learning*, Morgan Kaufmann Publishers, Inc. San Mateo, California, pp. 1–217.
OHTSUKI, M., MINATO, A. and OZAWA, S. (1999) Mathematical study of conceptualization in *n*-dimensional space, *Forma*, **14**, 303–307.
VALIANT, L. G. (1984) A theory of the learnable, *Communications of the ACM*, **27**, No. 11, 1134–1142.