AN INFINITE CLASS OF FUNCTIONS IDENTIFIABLE USING MINIMAL PROGRAMS IN ALL KOLMOGOROV NUMBERINGS

SANJAY JAIN

Department of Information Systems and Computer Science

National University of Singapore

Singapore 0511

Republic of Singapore

Email: sanjay@iscs.nus.sg

ABSTRACT

Identification of programs for computable functions from their graphs by algorithmic devices is a well studied problem in learning theory. Freivalds and Chen consider identification of 'minimal' and 'nearly minimal' programs for functions from their graphs. Freivalds showed that there exists a Gödel numbering in which only finite classes of functions can be identified using minimal programs. To address such problems, Freivalds later considered minimal identification in Kolmogorov Numberings. Kolmogorov numberings are in some sense optimal numberings and have some nice properties. Freivalds, showed that for every Kolmogorov numbering there exists an infinite class of functions which can be identified using minimal programs. Note that these infinite classes of functions may depend on the Kolmogorov numbering. It was left open whether there exists an infinite class of functions, $\mathcal{C}$, such that $\mathcal{C}$ can be identified using minimal programs in every Kolmogorov numbering. We show the existence of such a class.

Keywords: Inductive Inference; Minimal Identification; Kolmogorov Numbering.

1 Introduction

Let $N = \{0, 1, 2, \ldots\}$, the set of natural numbers. Let f be a function : $N \rightarrow N$. For any $n \in N$, we let $f[n]$ denote $\{(x, f(x)) \mid x < n\}$, the finite initial segment of f consisting of the first n data points in the graph of f. The quantifier '$\overset{\infty}{\forall}$' means 'for all but finitely many natural numbers'. Criteria of inference informally described below are formally defined in Section 3.

A *function learning machine* $\mathbf{M}$ is an algorithmic device which, on any input segment $f[n]$, returns either ? or a program, where, if it returns a program on a segment, it returns a program on all extensions of that segment. The output of $\mathbf{M}$ on $f[n]$ is denoted by $\mathbf{M}(f[n])$. If $\mathbf{M}(f[n])$ is a program, we think of that program as $\mathbf{M}$'s conjecture, based on the data $f[n]$, about how to compute all of f; $\mathbf{M}(f[n]) =?$ then represents the situation where $\mathbf{M}$ does not conjecture a program based on the data $f[n]$. The restriction that $\mathbf{M}$ must continue to conjecture programs once it has done so is essentially without loss of generality since a machine which hasn't had enough time to think of a new conjecture can be thought of re-outputting its previous conjecture.

As is by now well known, there are various senses in which $\mathbf{M}$ can be thought of as *successfully* learning or inferring a program for f. For $n \in N$, let $p_n = \mathbf{M}(f[n])$. The criterion of success known as $\mathbf{Ex}$-*identification* [10, 1, 3] requires that the sequence $p_0, p_1, p_2, \ldots$ contains a program *p, which computes f* , such that $(\stackrel{\infty}{\forall} i)[p_i = p]$. In this case one speaks of p as being the *final* program output by $\mathbf{M}$ on f.

Freivalds [6] and Chen [4, 5] studied the effect of requiring that the final hypothesis held by the learner in the above model be of (nearly) minimal size. Minimal identification of classes of functions depends on the acceptable programming system (acceptable numbering) chosen to interpret programs output by learning machines. Suppose ψ is a computable numbering (programming system). In $\mathbf{Min}_\psi\mathbf{Ex}$-identification criterion one requires, in addition to $\mathbf{Ex}$-identification of function (in the programming system ψ), that the final programs be of minimal size (see formal definitions in Section 3). We direct the reader to [6, 4, 5] for results dealing with minimal identification and its relationship with $\mathbf{Ex}$-identification.

Freivalds [6] showed that there exists a Gödel numbering ψ such that $\mathbf{Min}_\psi\mathbf{Ex}$ contains only finite classes of functions. This led Freivalds to consider minimal identification in Kolmogorov numberings. Kolmogorov numberings are computable numberings to which every computable numbering is reducible by a linearly bounded function. Freivalds [7, 8] showed that for every Kolmogorov numbering ψ, there exists an infinite class of functions in $\mathbf{Min}_\psi\mathbf{Ex}$. However he left open the question whether this result can be achieved using the same class for every Kolmogorov numbering. In other words, Freivalds left open the question whether there exists an infinite class, $\mathcal{C}$, of functions, such that for every

Kolmogorov numbering ψ, $\mathcal{C} \in \mathbf{Min}_\psi\mathbf{Ex}$? We show this to be true. In fact we prove a stronger result that there exists an infinite class $\mathcal{C}$ of functions such that $\mathcal{C} \in \mathbf{Min}_\psi\mathbf{FIN}$ for every Kolmogorov numbering ψ (for definition of **FIN**-identification see section 3).

We now proceed formally.

2 Notation

Recursion-theoretic concepts not explained below are treated in [12]. N denotes the set of natural numbers, $\{0, 1, 2, \ldots\}$. The symbols c, d, i, j, k, m, n, p and x, with or without decorations (decorations are subscripts, superscripts and the like), range over natural numbers unless otherwise specified. $\subseteq, \subset, \supseteq, \supset, \in$, denote subset, proper subset, superset, proper superset and element of relationship respectively. $\emptyset$ denotes the empty set. C, S, with or without decorations, range over subsets of N. We denote the cardinality of a set S by $\mathrm{card}(S)$. $\max(\ ), \min(\ )$ denote the maximum and minimum of a set, respectively. By convention $\max(\emptyset) = 0$ and $\min(\emptyset) = \infty$.

$\mathcal{R}$ denotes the set of all total recursive functions. h, f, g, with or without decorations, range over $\mathcal{R}$. $\mathcal{C}$, $\mathcal{H}$ and $\mathcal{S}$, with or without decorations, range over subsets of $\mathcal{R}$. $\downarrow$ denotes defined. $\uparrow$ denotes undefined.

A programming system (or computable numbering) is a (partial) computable function of two variables. We often drop the word 'computable' from 'computable numbering' in this paper, since we will be dealing with computable numberings only. We let ψ, β, η range over computable numberings (programming systems). Suppose $\psi(\cdot, \cdot)$ is a computable numbering. We often refer to the (partial) function $\lambda x.\psi(i, x)$ as ψ_i. ψ_i thus denotes the (partial) function computed by the i-th program in the numbering ψ.

An acceptable numbering is a computable numbering to which every other computable numbering is reducible via a recursive function. Thus if ψ is an acceptable numbering, then for all computable numberings η, there exists a recursive function h, such that $(\forall i)[\eta_i = \psi_{h(i)}]$. Acceptable numberings are also referred to as Gödel numberings. Kolmogorov numbering is an acceptable numbering to which every other computable numbering can be reduced via a linearly bounded function. Thus if ψ is a Kolmogorov numbering, then for all computable numberings η, there exists a recursive function h and a constant c, such that $(\forall i)[\eta_i = \psi_{h(i)} \ \wedge \ h(i) \leq \max(\{c * i, c\})]$.

For a function f, $\mathrm{MinProg}_\psi(f)$ denotes the minimal program for f, if any, in the ψ

programming system, i.e., $\mathrm{MinProg}_\psi(f) = \min(\{i \mid \psi_i = f\})$.

We let φ denote a standard acceptable programming system. φ_i thus denotes the partial recursive function computed by the i^{th} program in the standard acceptable programming system φ. We often refer to the i^{th} program as program i. Φ denotes an arbitrary fixed Blum complexity measure [2, 11] for the φ-system.

The quantifier '$\overset{\infty}{\exists}$' means 'there exist infinitely many'.

3 Learning Paradigms

For any partial function η and any natural number n such that, for each $x < n$, $\eta(x)\downarrow$, we let $\eta[n]$ denote the finite initial segment $\{(x, \eta(x)) \mid x < n\}$. Let INIT $= \{f[n] \mid f \in \mathcal{R} \wedge n \in N\}$. We let σ and τ, with or without decorations, range over INIT. $|\sigma|$ denotes the length of σ. Thus for example $|f[n]| = n$.

Definition 1 [10] A *learning machine* is an algorithmic device which computes a mapping from INIT into $N \cup \{?\}$ such that, if $\mathbf{M}(f[n]) \neq ?$, then $\mathbf{M}(f[n+1]) \neq ?$.

We let $\mathbf{M}$, with or without decorations, range over learning machines. In Definition 1 above, '?' denotes the situation when $\mathbf{M}$ outputs "no conjecture" on some $\sigma \in$ INIT.

In Definition 2 below we spell out what it means for a learning machine to converge in the limit.

Definition 2 Suppose $\mathbf{M}$ is a learning machine and f is a computable function. $\mathbf{M}(f)\downarrow$ (read: $\mathbf{M}(f)$ *converges*) just in case $(\exists i)(\overset{\infty}{\forall}\ n)\ [\mathbf{M}(f[n]) = i]$. If $\mathbf{M}(f)\downarrow$, then $\mathbf{M}(f)$ is defined = the unique i such that $(\overset{\infty}{\forall}\ n)[\mathbf{M}(f[n]) = i]$, otherwise we say that $\mathbf{M}(f)$ diverges (written: $\mathbf{M}(f)\uparrow$).

3.1 Explanatory Function Identification

We now formally define the criteria of inference considered in this paper.

Definition 3 [10, 3]

(a) A learning machine $\mathbf{M}$ is said to $\mathbf{Ex}$-*identify* f (written: $f \in \mathbf{Ex}(\mathbf{M})$) just in case $(\exists i \mid \varphi_i = f)(\overset{\infty}{\forall}\ n)[\mathbf{M}(f[n]) = i]$.

(b) $\mathbf{Ex} = \{\mathcal{C} \mid (\exists \mathbf{M})[\mathcal{C} \subseteq \mathbf{Ex}(\mathbf{M})]\}$.

3.2 Finite Function Identification

Definition 4

(a) A learning machine $\mathbf{M}$ is said to $\mathbf{FIN}$-*identify* f (written: $f \in \mathbf{FIN}(\mathbf{M})$) just in case $(\exists n, p \mid \varphi_p = f)[(\forall m < n)[\mathbf{M}(f[m]) =?] \ \wedge \ (\forall m \geq n)[\mathbf{M}(f[m]) = p]]$.

(b) $\mathbf{FIN} = \{\mathcal{C} \mid (\exists \mathbf{M})[\mathcal{C} \subseteq \mathbf{FIN}(\mathbf{M})]\}$.

3.3 Minimal Function Identification

We next consider identification by minimal programs. Minimal identification usually depends on the numbering system chosen.

Definition 5 [6] Suppose ψ is a numbering.

(a) $\mathbf{M}$ $\mathbf{Min}_\psi\mathbf{Ex}$-*identifies* f (written $f \in \mathbf{Min}_\psi\mathbf{Ex}(\mathbf{M})$) iff $\mathbf{M}(f)\downarrow = \mathrm{MinProg}_\psi(f)$.

(b) $\mathbf{Min}_\psi\mathbf{Ex} = \{\mathcal{C} \mid (\exists \mathbf{M})[\mathcal{C} \subseteq \mathbf{Min}_\psi\mathbf{Ex}(\mathbf{M})]\}$.

(c) $\mathbf{M}$ $\mathbf{Min}_\psi\mathbf{FIN}$-*identifies* f (written $f \in \mathbf{Min}_\psi\mathbf{FIN}(\mathbf{M})$) iff $(\exists n)[(\forall m < n)[\mathbf{M}(f[n]) = ?] \ \wedge \ (\forall m \geq n)[\mathbf{M}(f[n]) = \mathrm{MinProg}_\psi(f)]]$.

(d) $\mathbf{Min}_\psi\mathbf{FIN} = \{\mathcal{C} \mid (\exists \mathbf{M})[\mathcal{C} \subseteq \mathbf{Min}_\psi\mathbf{FIN}(\mathbf{M})]\}$.

4 Result

In this section we prove that there exists an infinite class of functions which can be identified using minimal programs in every Kolmogorov numbering. The theorem is proved using three lemmas. Let $\beta^0, \beta^1, \ldots$ denote a (non-effective) listing of all Kolmogorov numberings (note that we only need the listing for ease of reference to all the Kolmogorov numberings; thus non-effectiveness of the listing does not effect our result).

Intuitively Corollary 8 to Lemma 7 gives us a starting infinite class of functions with some nice properties. Lemma 6 allows us to generate a sequence of infinite classes, $\mathcal{C}_0 \supseteq \mathcal{C}_1 \supseteq \mathcal{C}_2 \supseteq \cdots$, such that $\mathcal{C}_{i+1} \in \mathbf{Min}_{\beta^i}\mathbf{FIN}$. Lemma 9 then allows us to construct the required class $\mathcal{C}$.

But first, we define a class $\mathcal{H}$ of functions and a useful predicate Good.

For all $j > 0$, let h_j be defined as follows.

$$h_j(x) = \begin{cases} j, & \text{if } x = 0; \\ 0, & \text{otherwise.} \end{cases}$$

Let $\mathcal{H} = \{h_j \mid j > 0\}$.

$$\text{Good}(\psi, i, c, d) \Leftrightarrow [[(\exists k \le c * i)[\psi_k = h_i]] \ \wedge \ [\text{card}(\{k \mid k \le c * i \ \wedge \ \psi_k(0) = i\}) \le d]]$$

Suppose $\mathcal{S} \subseteq \mathcal{H}$. Overloading the predicate Good, we say that

$$\text{Good}(\psi, \mathcal{S}) \Leftrightarrow (\exists c, d)(\forall j \mid h_j \in \mathcal{S})[\text{Good}(\psi, j, c, d)]$$

Intuitively, for an infinite $\mathcal{S}$, Good$(\psi, \mathcal{S})$, means that ψ satisfies some nice properties which allows one to finitely infer an infinite subset of $\mathcal{S}$ using minimal programs in the numbering ψ. This is the content of the following lemma.

Lemma 6 $(\forall \psi)(\forall \mathcal{S} \subseteq \mathcal{H} \mid \text{card}(\mathcal{S}) = \infty \wedge \text{Good}(\psi, \mathcal{S}))(\exists \mathcal{S}' \subseteq \mathcal{S})[\text{card}(\mathcal{S}') = \infty \wedge \mathcal{S}' \in \mathbf{Min}_\psi\mathbf{FIN}]$.

PROOF. Let ψ, $\mathcal{S}$ be as in the hypothesis of the lemma. Let c, d, be such that $(\forall j \mid h_j \in \mathcal{S})[\text{Good}(\psi, j, c, d)]$. Let $S_j = \{k \mid k \le c * j \ \wedge \ \psi_k(0) = j\}$. For $1 \le i \le d$, let $\mathbf{M}_i$ be defined as follows. The only program, if any, output by $\mathbf{M}_i$ on h_j, is the i-th program, if any, in a standard recursive enumeration of S_j. For all $h_j \in \mathcal{S}$, at least one of $\mathbf{M}_1, \ldots, \mathbf{M}_d$, $\mathbf{Min}_\psi\mathbf{FIN}$-identifies h_j (since Good(ψ, h_j, c, d)). Thus at least one of $\mathbf{M}_1, \ldots, \mathbf{M}_d$, $\mathbf{Min}_\psi\mathbf{FIN}$-identifies an infinite subset of $\mathcal{S}$. ∎

Lemma 7 ($\forall$ *Kolmogorov Numbering* $\psi)(\forall \epsilon > 0)(\exists c, d)(\forall j > 0)[\text{card}(\{k \mid 1 \le k \le j \ \wedge \ \neg\text{Good}(\psi, k, c, d)\}) < \epsilon * j]$.

PROOF. Suppose ψ, a Kolmogorov Numbering and $\epsilon > 0$ are given. Since $h_1, h_2, \ldots,$ is recursively enumerable, there exists a $c > 1$, such that $(\forall j > 0)(\exists k \le c * j)[\psi_k = h_j]$. Let $d = \lceil 2 * c/\epsilon \rceil$. Now the number of programs $\le c * j$, is $c * j + 1 \le 2 * c * j$ (for $j \ge 1$). Thus, it follows that $\text{card}(\{i \mid i \le j \ \wedge \ \text{card}(\{k \mid k \le c * j \ \wedge \ \psi_k(0) = i\}) > d\}) < \epsilon * j$. Thus $\text{card}(\{i \mid i \le j \ \wedge \ \text{card}(\{k \mid k \le c * i \ \wedge \ \psi_k(0) = i\}) > d\}) < \epsilon * j$. ∎

As a corollary we have

Corollary 8 *There exists an infinite* $\mathcal{S} \subseteq \mathcal{H}$ *such that* ($\forall$ *Kolmogorov Numbering* ψ)[Good($\psi, \mathcal{S}$)].

PROOF. Let $\epsilon_i = 2^{-i-3}$. For $i \in N$, let c_i, d_i, be c, d respectively as given by Lemma 7 for $\psi = \beta^i$ and $\epsilon = \epsilon_i$. Let $\mathcal{S}_i = \{h_j \mid \text{Good}(\beta^i, j, c_i, d_i)\}$. Let $\mathcal{S} = \bigcap_i \mathcal{S}_i$. Note that by Lemma 7, $(\forall i, j)[\text{card}(\{k \mid k \leq j \ \wedge \ h_k \notin \mathcal{S}_i\}) \leq j * 2^{-i-3}]$. It follows that $(\forall j)[\text{card}(\{k \mid k \leq j \ \wedge \ h_k \notin \mathcal{S}\}) \leq j * 2^{-2}]$. Thus $\mathcal{S}$ is infinite. ∎

Lemma 9 *Suppose* $\mathcal{C}_0, \mathcal{C}_1, \ldots$ *are given so that (1)* $\mathcal{C}_i \supseteq \mathcal{C}_{i+1}$ *and (2) each* $\mathcal{C}_i$ *is infinite. Then there exists an infinite* $\mathcal{C} \subseteq \mathcal{C}_0$*, such that* $(\forall i)[\text{card}(\mathcal{C} - \mathcal{C}_i) < \infty]$

PROOF. Let S_i be a subset of $\mathcal{C}_i$ with cardinality i. Let $\mathcal{C} = \bigcup_i S_i$. Clearly, $\mathcal{C}$ is infinite (since for each i, it contains a subset of size i of $\mathcal{C}_i$). Also, for each $i \in N$, since $[\bigcup_{j \geq i} S_j] \subseteq \mathcal{C}_i$, and $\text{card}(\bigcup_{j<i} S_j) < \infty$, we have that $\text{card}(\mathcal{C} - \mathcal{C}_i) < \infty$. Thus $\mathcal{C}$ satisfies the required properties. ∎

Theorem 10 *There exists an infinite class of functions* $\mathcal{C}$*, such that* ($\forall$ *Kolmogorov Numbering* ψ)[$\mathcal{C} \in \mathbf{Min}_\psi\mathbf{FIN}$].

PROOF. Let $\mathcal{C}_0 = \mathcal{S}$ as given by Corollary 8. For $i \in N$, let $\mathcal{C}_{i+1} = \mathcal{S}'$ as given by Lemma 6 by taking $\psi = \beta^i$ and $\mathcal{S} = \mathcal{C}_i$. It thus follows that

(a) $\mathcal{C}_0 \supseteq \mathcal{C}_1 \supseteq \mathcal{C}_2, \ldots$,

(b) each $\mathcal{C}_i$ is infinite and

(c) $\mathcal{C}_{i+1} \in \mathbf{Min}_{\beta^i}\mathbf{FIN}$.

For these $\mathcal{C}_i$'s let $\mathcal{C}$ be as given by Lemma 9. Now, for all i, since $\mathcal{C}_{i+1} \in \mathbf{Min}_{\beta^i}\mathbf{FIN}$, $\mathcal{C} - \mathcal{C}_{i+1}$ is finite and $(\forall f, g \in \mathcal{C})[f \neq g \Rightarrow f(0) \neq g(0)]$, it follows that $\mathcal{C} \in \mathbf{Min}_{\beta^i}\mathbf{FIN}$. ∎

5 Acknowledgements

I would like to thank Rusins Freivalds and Arun Sharma for helpful discussions on the subject.

References

[1] L. Blum and M. Blum. Toward a mathematical theory of inductive inference. *Information and Control*, 28:125–155, 1975.

[2] M. Blum. A machine independent theory of the complexity of recursive functions. *Journal of the ACM*, 14:322–336, 1967.

[3] J. Case and C. Smith. Comparison of identification criteria for machine inductive inference. *Theoretical Computer Science*, 25:193–220, 1983.

[4] K. Chen. *Tradeoffs in Machine Inductive Inference*. PhD thesis, SUNY at Buffalo, 1981.

[5] K. Chen. Tradeoffs in inductive inference of nearly minimal sized programs. *Information and Control*, 52:68–86, 1982.

[6] R. Freivalds. Minimal Gödel numbers and their identification in the limit. *Lecture Notes in Computer Science*, 32:219–225, 1975.

[7] R. Freivalds. Inductive inference of minimal programs. In M. Fulk and J. Case, editors, *Proceedings of the Third Annual Workshop on Computational Learning Theory*, pages 3–20. Morgan Kaufmann Publishers, Inc., August 1990.

[8] R. Freivalds. Inductive inference of recursive functions: Qualitative theory. In J. Barzdins and D. Bjorner, editors, *Baltic Computer Science. Lecture Notes in Computer Science 502*, pages 77–110. Springer-Verlag, 1991.

[9] R. Freivalds and S. Jain. Kolmogorov numberings and minimal identification. In *Proceedings of the Second European Conference on Computational Learning Theory*, March 1995. To Appear.

[10] E. M. Gold. Language identification in the limit. *Information and Control*, 10:447–474, 1967.

[11] J. Hopcroft and J. Ullman. *Introduction to Automata Theory Languages and Computation*. Addison-Wesley Publishing Company, 1979.

[12] H. Rogers. *Theory of Recursive Functions and Effective Computability.* McGraw Hill, New York, 1967. Reprinted by MIT Press, Cambridge, Massachusetts in 1987.