Algorithmic Dimensions via Learning Functions

Jack H. Lutz ✉ 🆔
Department of Computer Science, Iowa State University, Ames, IA, USA

Andrei N. Migunov ✉ 🆔
Department of Mathematics and Computer Science, Drake University, Des Moines, IA, USA

─── **Abstract** ───

We characterize the algorithmic dimensions (i.e., the lower and upper asymptotic densities of information) of infinite binary sequences in terms of the inability of learning functions having an algorithmic constraint to detect patterns in them. Our pattern detection criterion is a quantitative extension of the criterion that Zaffora Blando used to characterize the algorithmically random (i.e., Martin-Löf random) sequences. Our proof uses Lutz's and Mayordomo's respective characterizations of algorithmic dimension in terms of gales and Kolmogorov complexity.

1 Introduction

Algorithmic dimension was first formulated as a Σ_1^0 effectivization of classical Hausdorff dimension [12, 14].[1] The algorithmic dimension adim(Γ) of a set Γ of infinite binary sequences is, in fact, an upper bound of the Hausdorff dimension $\dim_H(\Gamma)$ of this set. Since algorithmic dimension has the *absolute stability* property that adim(Γ) is always the supremum of all adim$(\{X\})$ for $X \in \Gamma$, it was natural to define $\dim(X) = \text{adim}(\{X\})$ for all infinite binary sequences X and to investigate algorithmic dimension entirely in terms of the dimensions $\dim(X)$ of individual sequences X. Mayordomo [23] proved that the dimension $\dim(X)$, originally defined in terms of algorithmic betting strategies called *gales*, can also be characterized as the lower asymptotic density of the algorithmic information content of X. The more recent *point-to-set principle* [15] uses relativization to give the characterization

$$\dim_H(\Gamma) = \min_{A \subseteq \mathbb{N}} \sup_{X \in \Gamma} \dim^A(X)$$

of classical Hausdorff dimension. This principle has enabled several recent uses of computability theory to prove new classical theorems about Hausdorff dimension[21, 19, 20, 27, 18, 4, 5, 16, 17].[2]

Algorithmic dimension has the same Σ_1^0 "level of effectivization" as algorithmic randomness (also called "Martin-Löf randomness" [22] or, simply, "randomness"). In fact, every algorithmically random sequence X satisfies $\dim(X) = 1$, although the converse does not hold [14].

───────────

[1] Algorithmic dimension has also been called "constructive dimension", "effective Hausdorff dimension", and "effective dimension" by various authors.
[2] A theorem is "classical" here if its statement does not involve computability or related aspects of mathematical logic. Hence "new classical theorem" is not an oxymoron.

Computable learning theory, as initiated by Gold in 1967 [7, 25, 10], has been used to shed new light on randomness notions. Specifically, in 2008, Osherson and Weinstein [24] characterized two randomness notions, called weak 1-randomness and weak 2-randomness, for sequences X in terms of the inability of computable learning functions to detect patterns in X. Even more compellingly, Zaffora Blando [32] recently formulated a clever variant of Osherson and Weinstein's pattern detection criteria, called uniform weak detection, and used this to give an exact characterization of algorithmic (i.e., Martin-Löf) randomness.[3]

In this paper we introduce a quantitative version of Zaffora Blando's uniform weak detection criterion, called *s-learnability*, and we use this to characterize algorithmic dimension in terms of learning functions. Our main theorem says that, for every infinite binary sequence X, $\dim(X)$ is the infimum of all nonnegative real numbers s for which some learning function s-learns X. Our proof of this result uses methods of Osherson, Weinstein, and Zaffora Blando, together with martingale and Kolmogorov complexity techniques of Mayordomo [23]. We also characterize both the classical *packing dimension* $\dim_P$ [29, 30] and the *algorithmic strong dimension* $\mathrm{Dim}(X)$ [1] of a sequence in terms of learning functions. Along the way, we show that algorithmic randomness can also be characterized by specifically *polynomial-time* computable learning functions.

2   Preliminaries

Let $\mathbb{N}$ represent the natural numbers $\{0, 1, 2, ...\}$, $\mathbb{Q}$ the rationals, and $\mathbb{R}$ the reals. We will often use the extended naturals $\mathbb{N} \cup \{\infty\}$ and the extended reals $\mathbb{R} \cup \{\infty\}$. More often, we will refer to the respectively half open and closed intervals, $[0, \infty)$ and $[0, \infty]$.

We denote by $\{0, 1\}^*$ the set of all (finite) binary strings, and by $\{0, 1\}^\infty$ the set of all infinite binary sequences, which we call the *Cantor space* **C**. We denote the length in bits of a string or sequence w by $|w|$. The *empty string* is the unique string λ with $|\lambda| = 0$. If Z is an element of $\{0, 1\}^\infty$ or of $\{0, 1\}^*$, we write $Z \restriction n$ for the first n bits of Z if $|Z| \geq n$, and the value is undefined otherwise . Note that for all $n \in \mathbb{N}$, and all $Z \in \{0, 1\}^\infty$, $|Z \restriction n| = n$. We write $w \sqsubseteq Z$ if w is a *prefix* of Z, i.e., if $w = Z \restriction |w|$. We write $w \sqsubset Z$ if w is a *proper prefix* of Z, i.e., if w is a prefix of Z and $w \neq Z$.

For any string $w \in \{0, 1\}^*$, the *cylinder* at w is

$$C_w = \{Z \in \mathbf{C} \mid w \sqsubseteq Z\}.$$

If A is a set of strings, we denote the union of cylinders at those strings by $[\![A]\!] = \cup_{w \in A} C_w$. A (Borel) probability measure on **C** is a function $\mu : \{0, 1\}^* \to [0, 1]$ such that $\mu(\lambda) = 1$ and $\mu(w) = \mu(w0) + \mu(w1)$ for all $w \in \{0, 1\}^*$. Intuitively, $\mu(w)$ is the probability that $Z \in C_w$ when $Z \in \mathbf{C}$ is "chosen according to μ". In this sense, $\mu(w)$ is an abbreviation for $\mu(C_w)$. Standard methods [2] extend μ from cylinders to a σ-algebra $\mathcal{F}$ on **C**, so that $(\mathbf{C}, \mathcal{F}, \mu)$ is a probability space in the classical sense.

Most of our attention is on the uniform (Lebesgue) probability measure λ on **C** defined by $\lambda(w) = 2^{-|w|}$ for all $w \in \{0, 1\}^*$. We rely on context to distinguish Lebesgue measure from the empty string.

---

[3] Zaffora Blando's paper also characterized Schnorr randomness in terms of "computably uniform weak detection", but this is not germane to our work here.

▶ **Definition.** *A function* $f : \{0,1\}^* \to [0,\infty)$ *is lower semicomputable if there exists a computable function* $g : \{0,1\}^* \times \mathbb{N} \to \mathbb{Q} \cap [0,\infty)$ *such that for all* $w \in \{0,1\}^*, t \in \mathbb{N}$,

$$g(w,t) \leq g(w,t+1) \leq f(w)$$

and

$$\lim_{t \to \infty} g(w,t) = f(w).$$

We will say that a set S of real numbers is *uniformly left computably enumerable (uniformly left c.e.)* if there exists a lower-semicomputable function whose range is S.

3 Algorithmic randomness via learning functions

The algorithmic randomness of a sequence was originally defined in [22] in terms of algorithmic measure theory. In this view, an algorithmically random sequence is one which belongs to every algorithmically definable measure one set. Martin-Löf shows that all algorithmically nonrandom sequences belong to one universal algorithmically measure-zero set.

▶ **Definition** ([22]). *If* μ *is a probability measure on* $\mathbf{C}$, *then we say a set* X *has* algorithmic μ-measure zero *if there exists a computable function* $g : \mathbb{N} \times \mathbb{N} \to \{0,1\}^*$ *such that: for every* $k \in \mathbb{N}$,

$$X \subseteq \bigcup_{n=0}^{\infty} C_{g(k,n)}$$

and

$$\sum_{n=0}^{\infty} \mu(C_{g(k,n)}) \leq 2^{-k}.$$

▶ **Definition** ([22]). *We say a sequence* S *is* Martin-Löf μ-nonrandom *if* $\{S\}$ *has algorithmic* μ-*measure zero, and* Martin-Löf μ-random *otherwise.*

When the probability measure μ is the Lebesgue measure λ defined in section 2, we omit it from the terminology in the preceding two definitions. Randomness can also be characterized using gambling strategies called *gales*.

▶ **Definition** ([14]). *For* $s \in [0,\infty)$, *a* μ-*s*-gale *is a function* $d : \{0,1\}^* \to [0,\infty)$ *that satisfies the condition that*

$$d(w)\mu(w)^s = d(w0)\mu(w0)^s + d(w1)\mu(w1)^s,$$

for every $w \in \{0,1\}^*$.

Sometimes, when the probability distribution μ is clear from context, we will refer simply to *s*-gales. A *martingale* is a 1-gale. If not stated explicitly otherwise, we assume that the *initial capital* of a gale is $d(\lambda) = 1$.

▶ **Definition.** *A* μ-*s-gale* d succeeds *on a set* Γ *of sequences if*

$$\limsup_{n \to \infty} d(X \upharpoonright n) = \infty$$

for every sequence $X \in \Gamma$.

Ville shows the following:

▶ **Theorem 1** ([31]). *Let $\lambda(E)$ denote the Lebesgue measure of a set $E \subseteq \{0,1\}^\infty$. The following are equivalent:*
(1) $\lambda(E) = 0$
(2) *There exists a martingale $d : \{0,1\}^* \to [0,\infty)$ that succeeds on E.*

Schnorr effectivizes Ville's theorem as follows:

▶ **Theorem 2** ([26]). *A set of sequences Γ is Martin-Löf random if and only if there exists no lower-semicomputable martingale that succeeds on Γ.*

We will revisit gales in a later section, when we discuss dimension.

▶ **Definition** ([25]). *A function $l : \{0,1\}^* \to \{\text{YES}, \text{NO}\}$ is called a* learning function.

YES and NO are simply aliases for 1 and 0, respectively.

One can impose resource bounds on learning functions or on properties of these functions such as their average answers "along" a string w. l may be computable (in which case we call it a computable learning function or *CLF*), or l may have lower-semicomputable averages at all points in $\{0,1\}^*$, or any number of other resource restrictions.

▶ **Definition** ([32]). *A learning function l is said to* uniformly weakly detect *that a sequence $X \in \{0,1\}^\infty$ is patterned if and only if*
1. $l(X \restriction m) = \text{YES}$ *for infinitely many $m \in \mathbb{N}$, and*
2. $\lambda(\{Y \in \{0,1\}^\infty \mid \#\{m \in \mathbb{N} \mid l(Y \restriction m) = \text{YES}\} \geq n\}) \leq 2^{-n}$
for all $n \in \mathbb{N}$.

Zaffora Blando shows that computable learning functions and uniform weak detectability characterise Martin-Löf randomness:

▶ **Theorem 3** ([32]). *A sequence $X \in \{0,1\}^\infty$ is Martin-Löf random if and only if there is no computable learning function that uniformly weakly detects that X is patterned.*

▶ **Theorem 4.** *The following are equivalent:*
1. *There exists a computable learning function that uniformly weakly detects that X is patterned.*
2. *There exists a polynomial-time computable learning function that uniformly weakly detects that X is patterned.*

Proof. $(2) \implies (1)$ is immediate.

To see that $(1) \implies (2)$:

Let l be a computable learning function which uniformly weakly detects that X is patterned. Let $M_l(w)$ be a TM which computes $l(w)$. Algorithm 1 specifies a learning function with the following properties:

$\hat{l} = \text{YES}$ if there exist $w' \sqsubset w$ and $t_{w'} \in \mathbb{N}$ such that all of the following hold:
1. $|w| = |w'| + t_{w'}$
2. $l(w') = \text{YES}$
3. $t_{w'} = \min\{t \mid M_l(w') = \text{YES} \text{ after } t \text{ steps}\}$,
and $\hat{l} = \text{NO}$ otherwise.

If l says YES infinitely often on X, then there are infinitely many $w', t_{w'}$ where $M_l(w') = \text{YES}$ after exactly $t_{w'}$ time steps. Thus there are infinitely many w with $|w| = |w'| + t_{w'}$ where $\hat{l}$ says YES. Thus, $\hat{l}$ says YES infinitely often on X.

For all $Y \in \{0,1\}^\infty$, the number of YES given by $\hat{l}$ along Y remains the same as the number given by l, all such answers being "delayed" until later. Thus, the measure condition is satisfied. Thus, $\hat{l}$ is a learning function which uniformly weakly detects that X is patterned.

■ **Algorithm 1** $M_{\hat{l}}$.

1: Input w:
2: **for all** $w' \sqsubset w$ **do**
3:    Run $M_l(w')$ for exactly $|w| - |w'|$ steps.
4:    **if** $M_l(w')$ prints YES for the first time after exactly $|w| - |w'|$ steps **then**
5:       **return** YES
6:    **else**
7:       Continue
8:    **end if**
9: **end for**
10: **return** NO

$M_{\hat{l}}$ always halts and terminates in time polynomial in $|w|$. Thus, $\hat{l}$ is polynomial-time computable. ◀

As a result, one can characterize algorithmic randomness in terms of polynomial-time computable learning functions:

▶ **Corollary 5.** *A sequence $X \in \{0,1\}^\infty$ is Martin-Löf random if and only if there is no polynomial-time computable learning function that uniformly weakly detects that X is patterned.*

We also note a useful fact about uniform weak detectability:

▶ **Observation 6.** *If l_1 uniformly weakly detects that Γ_1 is patterned, and l_2 uniformly weakly detects that Γ_2 is patterned, then there exists a learning function l_3 which uniformly weakly detects that $\Gamma_1 \cup \Gamma_2$ is patterned. This transformation preserves computability.*

Proof. Define l_3 by:

$l_3(v) = $ YES if either $l_1(v) = $ YES or $l_2(v) = $ YES, unless for all $v' \sqsubset v$ $l_1(v) = $ NO or for all $v' \sqsubset v$ $l_2(v) = $ NO.

That is, l_3 says YES whenever either l_1 or l_2 would say YES, except for the first time for each.

It is easy to verify that l_3 says YES infinitely often on every $X \in \Gamma_1 \cup \Gamma_2$ and that the measure property is satisfied. It is also easy to show that if l_1 and l_2 are computable, then l_3 is as well. ◀

As a result, uniform weak detectability by computable learning functions is closed under finite unions.

4 Classical and algorithmic dimensions

Next, we review the definitions of classical and algorithmic dimensions.

We say a set A *covers* a set of sequences Γ if for every $X \in \Gamma$ there is some $w \in A$, $w \sqsubseteq X$. Let $k \in \mathbb{N}$, and let $\mathcal{A}_k = \{A \mid A \text{ is a prefix set and } \forall x \in A, |x| \geq k\}$. Let

$$\mathcal{A}_k(\Gamma) = \{A \in \mathcal{A}_k \mid A \text{ covers } \Gamma\},$$

and let

$$H_k^s(\Gamma) = \inf_{A \in \mathcal{A}_k(\Gamma)} \sum_{w \in A} 2^{-s|w|}.$$

Note that this infimum is taken only over sets of cylinders, not over all possible covers. For that reason, the following function is a proxy for - and is within a constant multiplicative factor of - what is know as the *s-dimensional Hausdorff outer measure*, in which all covers are considered.

▶ **Definition** ([8]). $H^s(\Gamma) = \lim_{k \to \infty} H_k^s(\Gamma)$.

For any set Γ, there exists some $s \in [0, \infty)$ such that for every $a < s < b$,
1. $H^a(\Gamma) = \infty$, and
2. $H^b(\Gamma) = 0$.

The real number s is the Hausdorff dimension of Γ:

▶ **Definition** ([8]). *The* Hausdorff dimension *of a set* $\Gamma \subseteq \{0, 1\}^\infty$ *is*

$$\dim_H(\Gamma) = \inf\{s \in [0, \infty) \mid H^s(\Gamma) = 0\}.$$

Hausdorff dimension can be characterized in terms of gales:

▶ **Theorem 7** ([13]).

$$\dim_H(\Gamma) = \inf\{s \in [0, \infty) \mid \text{ there exists an s-gale that succeeds on } \Gamma\}.$$

Lutz also showed that by effectivizing gales[4] at various levels, one can obtain various *effective* dimension notions, including the *algorithmic dimension*:

▶ **Definition** ([14]). *The* algorithmic dimension *of a set* $\Gamma \subseteq \{0, 1\}^\infty$ *is*

$$\mathrm{adim}(\Gamma) = \inf\{s \in [0, \infty) \mid \text{ there exists}$$

a lower semicomputable s-gale that succeeds on $\Gamma\}$.

We also note the following theorem:

▶ **Theorem 8** ([14]). *Let* μ *be a probability measure on* $\{0, 1\}^\infty$, *let* $s, s' \in [0, \infty)$, *and let* $d, d' : \{0, 1\}^* \to [0, \infty)$. *Assume that*

$$d(w)\mu(w)^s = d'(w)\mu(w)^{s'}$$

for all $w \in \{0, 1\}^*$. *Then,* d *is a* μ-s-gale *if and only if* d' *is a* μ-s'-gale.

A corollary of this theorem, which we will make use of in the main proof of the next section, is the following:

▶ **Proposition 9.** *If* d *is an s-gale that succeeds on* X *then there exists a martingale* d' *that succeeds on* X *against order* $h(w) = 2^{(1-s)|w|}$. *That is,*

$$\limsup_{n \to \infty} \frac{d(X \upharpoonright n)}{h(n)} = \infty.$$

---

[4] In Lutz' original proof, *supergales* are used. These satisfy the gale definition with $\leq$ in place of equality. [9] shows that gales suffice for the characterization of algorithmic dimension.

Another important classical dimension notion is the packing dimension [30, 29]. As with the Hausdorff dimension, it is easier to define it in terms of gales than in terms of its original conception via coverings. First we define a notion of *strong success* for gales:

▶ **Definition** ([1]). *A μ-s-gale d succeeds strongly* on a set Γ of sequences if

$$\liminf_{n \to \infty} d(X \restriction n) = \infty$$

for every sequence $X \in \Gamma$.

▶ **Definition** ([1]). *The* packing dimension *of a set $\Gamma \subseteq \{0,1\}^\infty$ is*

$$\dim_P(\Gamma) = \inf\{s \in [0, \infty) | \text{ there exists an s-gale that succeeds strongly on } \Gamma\}.$$

The packing dimension can be effectivized as follows:

▶ **Definition** ([1]). *The* algorithmic packing dimension *or* algorithmic strong dimension *of a set $\Gamma \subseteq \{0,1\}^\infty$ is*

$$\mathrm{aDim}(\Gamma) = \inf\{s \in [0, \infty) | \text{ there exists a lower semi-computable s-gale}$$

$$\text{that succeeds strongly on } \Gamma\}.$$

We use the above notations adim and aDim when describing the algorithmic dimensions of sets in order to distinguish these from their classical counterparts. When applied to individual sequences, we often use dim and Dim, respectively, as there is no ambiguity.

5 Hausdorff dimension via learning functions

Learning functions can be used to characterize the classical (Hausdorff) dimension. Once such a characterization is in place, one can impose further restrictions on the computability of learning functions and thereby use the notion of learning to characterize dimension notions at various levels of effectivity.

We refine the definition of uniform weak detectability, by adding a requirement on the frequency of YES answers, in order to characterize dimension. Recall that we identify YES with 1 and NO with 0, thus the sum in the following definition is well-defined.

▶ **Definition.** *If l is a learning function, the* path average of l up to w *is denoted*

$$\mathrm{AVG}_l(w) = \frac{\sum_{i=0}^{i=|w|} l(w \restriction i)}{|w|}.$$

Let Δ be a computability restriction such as "lower semi-computable", "computable", or "all" (the absence of a restriction).

▶ **Definition.** *A sequence X is (Δ)-s-learnable if and only if there exists a function $l : \{0,1\}^* \to \{\mathrm{YES}, \mathrm{NO}\}$ such that the following three conditions are satisfied.*
1. *For every $w \in \{0,1\}^*$, the path averages $\mathrm{AVG}_l(w)$ are uniformly Δ-computable (in w).*
2. *For all $n \in \mathbb{N}$,*

$$\lambda(\{Y \in \{0,1\}^\infty \mid \#\{i \mid l(Y \restriction i) = \mathrm{YES}\} \geq n)\}) \leq 2^{-n}.$$

3. $\limsup_{n \to \infty} \mathrm{AVG}_l(X \restriction n) \geq 1 - s.$

We say that a sequence X is *strongly (Δ)-s-learnable* if it satisfies (1) and (2) above, and satisfies condition (3) with a $\liminf$ rather than a $\limsup$.

Specifically, we will say that *algorithmic s-learnability* corresponds to $\Delta = \Sigma_1^0$, *computable s-learnability* corresponds to $\Delta = \Delta_1^0$ and *s-learnability as such* corresponds to no resource restriction (in other words, any learning functions at all can be used). Note that the restriction on computability is not applied to the learning function itself, but to its path averages on the elements of $\{0, 1\}^*$.

We often refer to a learning function which satisfies these properties as an *s-learner*, and we say it *s-learns* a sequence or set of sequences.

This definition is in the spirit of [28], emphasizing learnability criteria based on frequencies of YES answers, rather than restrictions on the measure conditions.

A function $l : \{0, 1\}^* \to \{\text{NO}, \text{YES}\}$ is associated to a transformation $\hat{l} : \{0, 1\}^\infty \to \{\text{NO}, \text{YES}\}^\infty$ defined by $\hat{l}(Y) = X$, where $X[n] = l(Y \upharpoonright n)$. In this sense, each sequence x is transformed into a sequence consisting of YES and NO "bits". For any set S, and function $f : \{0, 1\}^\infty \to \{0, 1\}^\infty$, $f^{-1}[S]$ is the set of all X with $f(X) \in S$. Define functions $\hat{l}$ on finite strings analogously.

▶ **Observation 10.** *For every learner l and every $Y \in \{0, 1\}^\infty$, either l s-learns every $X \in \hat{l}^{-1}[Y]$ or none of them.*

This is immediate from the success criterion. s-learning a sequence is purely a matter of the asymptotic frequency with which some learner says YES on prefixes of that sequence, assuming the measure condition is satisfied by l.

Closure under finite unions also holds for s-learnability:

▶ **Observation 11.** *If l_1 s-learns Γ_1 and l_2 s-learns Γ_2, then there exists a learning function l_3 which s-learns $\Gamma_1 \cup \Gamma_2$.*

Proof. The idea is the same as in Observation 6. Define l_3 by:

$l_3(v) = \text{YES}$ if either $l_1(v) = \text{YES}$ or $l_2(v) = \text{YES}$, unless it's the first time that either l_1 or l_2 has said YES on any $v' \sqsubseteq v$.

Note that $\limsup_{n \to \infty} \frac{\sum^n l(X \upharpoonright i)}{n} \geq (1 - s)$ implies that $\limsup_{n \to \infty} \frac{\sum^n l(X \upharpoonright i)}{n} - \frac{k}{n} \geq (1 - s)$ for any fixed k.   ◀

We will now show that s-learning characterizes Hausdorff dimension.

Let $\mathscr{G}(\Gamma) = \{s \in (0, \infty) \mid \text{there exists a learning function } l \text{ which } s\text{-learns every } X \in \Gamma\}$. The following theorem is a characterization of Hausdorff dimension in terms of learning functions.

▶ **Theorem 12.** *For all $\Gamma \subseteq \{0, 1\}^\infty$,*

$$\dim_H(X) = \inf \mathscr{G}(\Gamma).$$

Proof. Assume $\dim_H(\Gamma) \leq s$. Then for all $s' > s$ there exists an s'-gale which succeeds on Γ, and by Proposition 9 there exists a martingale d which, for every $X \in \Gamma$ succeeds on X against order $h(w) = 2^{(1-s')|w|}$. That is, d doubles its money asymptotically $1 - s'$ share of the time. Let

$$\gamma_d(w) = \begin{cases} \text{YES} & \text{if there exists } w' \sqsubseteq w \text{ such that } d(w) \geq 2 \cdot d(w') \text{ and} \\ & \forall \hat{w} \text{ satisfying } w' \sqsubset \hat{w} \sqsubset w, \gamma_d(\hat{w}) = \text{NO} \\ \text{NO} & \text{otherwise} \end{cases} \tag{5.1}$$

γ_d says YES every time d doubles its money (attains a new 2^k value). The set of sequences on which γ_d says YES infinitely often is the same as the set of sequences on which d wins unbounded money *against h*, thus it has measure zero. Thus for any $X \in \Gamma$ there must be infinitely many $X \upharpoonright n$ such that $\gamma_d(X \upharpoonright n)$ has YES density at least $(1 - s')$ at $X \upharpoonright n$.

Let $n \in \mathbb{N}$ and let $B_n = \{w \mid d(w) \geq 2^n$ and $\forall v \sqsubseteq w, d(v) < 2^n\}$. B_n is the (prefix-)set of all w at which d has accumulated 2^n value for "the first time". By the Kolmogorov inequality ([31]), $\lambda(\cup_{w \in B_n} C_w) \leq 2^{-n}$. For all n,

$$A_n^{\gamma_d} = \{Y \mid \#\{m \mid \gamma_d(Y \upharpoonright m) = \text{YES}\} \geq n\}$$
$$\subseteq \{Y \mid \exists k \ d(Y \upharpoonright k) \geq 2^n\}$$
$$= \cup_{w \in B_n} C_w,$$

Thus, for all n, $\lambda(A_n^{\gamma_d}) \leq \lambda(\cup_{w \in B_n} C_w) \leq 2^{-n}$. Thus γ_d s'-learns every $X \in \Gamma$.

In the other direction, suppose that for all $s' > s$ there is a learning function which s'-learns every $X \in \Gamma$. Let l be such a learning function. We show $\dim_H(\Gamma) \leq s$. Let k, r be any integers. Let

$$\hat{A}_k = \{w \mid \# \text{YES}(w) \geq r + (1 - s')|w| \text{ and } \forall v \sqsubset w, v \notin \hat{A}_k\}.$$

$\hat{A}_k \in \mathcal{A}_k(\Gamma)$ because for all $w \in \hat{A}_k$, $|w| > k$, and $\hat{A}_k$ is a prefix set.
For every n,

$$\lambda(\{Y \mid \# \text{YES}(Y) \geq r + (1 - s')n\}) \leq 2^{-r - (1 - s')n}.$$

Thus, the number of strings of length exactly k which can have at least $r + (1 - s')k$ YES answers is at most

$$\frac{2^{-r - (1 - s')k}}{2^{-k}} = 2^{-r + s'k}.$$

$$H_k^{s'}(\Gamma) \leq \sum_{w \in \hat{A}_k} 2^{-s'|w|} \leq \sum_{n=k}^{\infty} |\hat{A}_k^{=n}| 2^{-s'n} \leq 2^{-r + s'k} 2^{-s'k} = 2^{-r},$$

where the third inequality is due to the Kraft inequality [11, 3], because $\hat{A}_k$ is a prefix set.
Thus

$$H^{s'}(\Gamma) = \lim_{k \to \infty} \inf_{A \in \mathcal{A}_k(\Gamma)} \sum_{w \in A} 2^{-s|w|}$$
$$\leq \lim_{k \to \infty} \sum_{w \in \hat{A}_k} 2^{-s|w|}$$
$$= 0.$$

Thus $\dim_H(\Gamma) \leq s$. ◄

6 Algorithmic dimension via learning functions

In this section, we show that algorithmic s-learning characterizes algorithmic dimension.

Recall that a learning function *algorithmically s-learns* a set $\Gamma \subseteq \{0, 1\}^\infty$ if it satisfies the definition of s-learning with $\Delta = \Sigma_1^0$. In other words, we require that the path-averages of l on all strings w are uniformly lower semicomputable real numbers.

Before we discuss the proof that algorithmic s-learnability characterizes algorithmic dimension, we note why it is at least *seemingly difficult* to directly lower semi-compute the learning function described in Theorem 12. For one, a suitable notion of lower semi-computability is hard to establish for functions mapping to $\{0, 1\}$. Secondly, the learning function γ_d in the proof of Theorem 12 relies on being able to place a YES answer every time the "underlying" martingale doubles its money for the first time. That is, every time it achieves a new value 2^k for the first time along some sequence. Though checking whether this happens *eventually* is lower semicomputable, checking whether it has happened at a particular w for the *first time* is not. Instead, we can simply require that path-averages be lower semicomputable. Thus, the conditions of algorithmic s-learnability coincide with the existence of lower semi-computable martingales succeeding against exponential orders.

For a given martingale d, we define the learning function γ_d as in Theorem 12. Note that if d is lower semi-computable with computable witness $\hat{d}$, then the path averages of γ_d are lower semi-computable uniformly in w via the computable witness below (Algorithm 2).

■ **Algorithm 2** AVG_l.

1: Input w, k:
2: Compute all the values $\hat{d}(\lambda, k), \ldots, \hat{d}(w, k)$.
3: Compute $m_{w,k} = \max_{w' \sqsubseteq w} \left\{ \log_2 \left(\hat{d}(w', k) \right) \right\}$
4: **return** $\frac{m_{w,k}}{|w|}$

As a result, the set

$$T = \{w \mid \mathrm{AVG}_{\gamma_d}(w) \geq 1 - s\}$$

is computably enumerable, and so are all of the slices

$$T_m = \{w \mid \mathrm{AVG}_{\gamma_d}(w) \geq 1 - s \text{ and } |w| = m\}.$$

Let $\mathscr{G}_{\mathrm{alg}}(\Gamma) = \{s \in [0, \infty) \mid \text{there exists a learning function } l \text{ which algorithmically } s\text{-learns}$ every $X \in \Gamma\}$.

We now prove our main theorem.

▶ **Theorem 13.** *For all $\Gamma \subseteq \{0, 1\}^\infty$,*

$$\mathrm{adim}(\Gamma) = \inf \mathscr{G}_{\mathrm{alg}}(\Gamma).$$

Proof. Let $s = \inf \mathscr{G}_{\mathrm{alg}}(\Gamma)$. We will show $\dim(\Gamma) \leq s$.

Let $s' > s$ and let l be a learning function that algorithmically s'-learns every $X \in \Gamma$.

[6] Theorem 13.3.4 ([23], Theorem 3.1) states that $\dim(X) = \liminf_n \frac{C(X \restriction n)}{n}$ where C is the *plain Kolmogorov complexity.*

Let $T = \cup T_n$ where $T_n = \{w \mid |w| = n \text{ and } \mathrm{AVG}_l(w) \geq 1 - s\}$. T is computably enumerable due to the first condition of s'-learnability.

For each n, $\lambda(\cup_{w \in T_n} C_w) \leq 2^{-(1-s')n}$ as a result of the measure condition of s'-learnability, since every sequence Y with at least $n(1-s)$ YES answers at length n is in the set of sequences which have at least $n(1-s)$ YES answers total, and the measure of the latter set is at most $2^{-(1-s')n}$. Thus, since each $w \in T_n$ has measure 2^{-n} and all are disjoint, we have $|T_n| \leq 2^{s'n}$. As a result, in plain Kolmogorov complexity terms, we only need to supply at most $s'n$ bits to identify an element of T living in T_n (we get n "for free" as long as we supply *exactly* $s'n$ bits, and use them to identify the slice T_n of T).

This means that for all $w \in T$, $C(w) \leq s'|w|$. Note that for every $X \in \Gamma$, there are infinitely many n so that $X \upharpoonright n \in T$.

It follows, then, that for every $X \in \Gamma$,

$$\dim(X) = \liminf_n \frac{C(X \upharpoonright n)}{n} \leq \frac{s'n}{n} = s',$$

and thus - since $s' > s$ was arbitrary - $\dim(\Gamma) \leq s$.

In the other direction, we show that if $\dim(\Gamma) \leq s$, then for every $s' > s$, Γ is algorithmically s'-learnable.

Again, let $s' > s$ and let d be a martingale which succeeds against order $2^{(1-s')n}$ on every $X \in \Gamma$. Define the learning function γ_d, as in Theorem 12 (Eq. 5.1), to say YES every time d doubles its money (attains a new 2^k value) for the first time along each path.

We have established that the values $\text{AVG}_{\gamma_d}(w)$ are uniformly lower-semicomputable when d is lower-semicomputable. We also know from the proof of Theorem 12 that γ_d satisfies the measure condition and that $\limsup_{n \to \infty} \text{AVG}(l, X \upharpoonright n) \geq 1 - s'$, for every $X \in \Gamma$. ◄

7 Strong algorithmic dimension via learning functions

In this section, we characterize the packing dimension as well as the strong algorithmic dimension of a set of binary sequences in terms of strong s-learning and strong algorithmic s-learning, respectively.

For any $\Gamma \subseteq \{0,1\}^\infty$, let $\Gamma \upharpoonright n = \{w \mid |w| = n$ and w is a prefix of some $v \in \Gamma\}$.

▶ **Definition** ([6]). *For any $\Gamma \subseteq \{0,1\}^\infty$, let the* upper box-counting dimension *of Γ be*

$$\overline{\dim_B(\Gamma)} = \limsup_n \frac{\log |\Gamma \upharpoonright n|}{n}.$$

▶ **Theorem 14** ([1]). *For every $X \in \{0,1\}^\infty$,*

$$\text{Dim}(X) = \limsup_n \frac{C(X \upharpoonright n)}{n}.$$

Let $\mathscr{G}_{str}(\Gamma) = \{s \in [0, \infty) \mid$ there exists a learning function f which strongly s-learns every $X \in \Gamma\}$.

▶ **Theorem 15.** *Let $\Gamma \subseteq \{0,1\}^\infty$. Then,*

$$\dim_P(\Gamma) = \inf \mathscr{G}_{str}(\Gamma).$$

Proof. First, we show that if, for arbitrary $s' > s$, l is a learning function that strongly s'-succeeds on Γ, then $\dim_P(\Gamma) \leq s$.

The proof is much the same as [1] and [4] Theorem 13.11.9, except we start with a learning function instead of a gale. Assume the hypothesis. Let $\hat{s} > s'$ be arbitrary.

Let

$$T_n = \{w \mid \text{AVG}_l(w) \geq 1 - \hat{s} \text{ and } |w| = n\}$$

be the set of strings of length n on which l achieves the requisite density. Then, for all $X \in \Gamma$, and for all but finitely many $n \in \mathbb{N}$, $X \in \cup_{w \in T_n} C_w$, i.e.

$$\Gamma \subseteq \cup_i \cap_{j \geq i} [\![T_j]\!].$$

Let $Z_n = \cap_{m \geq n} [\![T_m]\!]$. Then, $X \in Z_n$ means that after the nth prefix, all remaining prefixes of X have density at least $1 - \hat{s}$.

It suffices - by the countable stability of $\dim_P$ and the fact that $\dim_P \leq \overline{\dim_B}$ (see [6] 13.11.3) - to show that $\overline{\dim_B}(Z_n) \leq \hat{s}$, for all n. While the original proof uses Kolmogorov's inequality, we are not dealing with a gale. We have shown in the proof of Theorem 13 that if l is an $\hat{s}$-learner, then the set T_n satisfies $|T_n| \leq 2^{\hat{s}n}$.

Then,

$$\overline{\dim_B(Z_i)} = \limsup_n \frac{\log |Z_i \upharpoonright n|}{n} \leq \limsup_n \frac{\log |T_n|}{n} \leq \hat{s}.$$

Since $\hat{s} > s' > s$ are arbitrary, $\dim_P(\Gamma) \leq \overline{\dim_B}(\Gamma) = \overline{\dim_B}(\cup Z_i) \leq s$.

In the other direction, simply note that if $\dim_P(\Gamma) \leq s$ then for every $s' > s$ there exists a martingale $d_{s'}$ which strongly s'-succeeds against order $2^{(1-s')n}$ on every $X \in \Gamma$, and the frequency with which the martingale doubles its money is always eventually bounded below. Thus, a learner defined as in Eq. 5.1 will have the desired YES density on the same sequences.   ◀

Let $\mathscr{G}_{\mathrm{alg}}^{str}(\Gamma) = \{s \in [0, \infty) \mid \text{there exists a learning function } l \text{ which strongly algorithmically } s\text{-learns every } X \in \Gamma\}$.

▶ **Theorem 16.** *Let $\Gamma \subseteq \{0,1\}^{\infty}$. Then,*

$$\mathrm{aDim}(\Gamma) = \inf \mathscr{G}_{\mathrm{alg}}^{str}(\Gamma).$$

Proof. Much like the gale characterization of aDim resembles the gale characterization of adim with minor changes (see [1] and [6], Corollary 13.11.12), this proof closely resembles the proof of Theorem 13.

We replace the $\liminf$ with a $\limsup$ in order to apply Theorem 14, and we replace the observation that infinitely many n satisfy $X \upharpoonright n \in T$ with "all but finitely many."

In the other direction, we assume there exists a lower semi-computable martingale $d_{s'}$ which succeeds strongly against order $2^{(1-s')n}$ on every $X \in \Gamma$. A learning function defined in the same way as before (Eq. 5.1) will have the requisite density of YES answers, and will also have uniformly lower-semicomputable averages at each $w \in \{0,1\}^*$, as established in Section 6.   ◀

───── **References** ─────

1   Krishna B. Athreya, John M. Hitchcock, Jack H. Lutz, and Elvira Mayordomo. Effective strong dimension in algorithmic information and computational complexity. *SIAM journal on computing*, 37(3):671–705, 2007.

2   Patrick Billingsley. *Probability and measure*. Wiley-Interscience, 1995.

3   Thomas R. Cover and Joy A. Thomas. *Elements of Information Theory*. John Wiley & Sons, Inc., 2006.

4   Rod Downey and Denis R Hirschfeldt. Algorithmic randomness. *Communications of the ACM*, 62(5):70–80, 2019.

5   Rod Downey and Denis R Hirschfeldt. Computability and randomness. *Notices of the American Mathematical Society*, 66(7):1001–1012, 2019.

6   Rodney G. Downey and Denis R. Hirschfeldt. *Algorithmic Randomness and Complexity*. Springer Science & Business Media, 2010.

7   E Mark Gold. Language identification in the limit. *Information and control*, 10(5):447–474, 1967.

8   F Hausdorff. Dimension und äußeres maß. *Mathematische Annalen*, 79:157–179, 1919.

9   John M Hitchcock. Gales suffice for constructive dimension. *Information Processing Letters*, 86(1):9–12, 2003.

10   Sanjay Jain, Daniel Osherson, James S Royer, Arun Sharma, et al. *Systems that Learn: An Introduction to Learning Theory*. MIT press, 1999.

11   Leon G Kraft. A device for quantizing, grouping and coding amplitude modified pulses. *Master's thesis, Cambridge, MA*, 1949.

12   Jack H Lutz. Gales and the constructive dimension of individual sequences. In *International Colloquium on Automata, Languages, and Programming*, pages 902–913. Springer, 2000.

13   Jack H Lutz. Dimension in complexity classes. *SIAM Journal on Computing*, 32(5):1236–1259, 2003.

14   Jack H. Lutz. The dimensions of individual strings and sequences. *Information and Computation*, 187(1):49–79, 2003. `doi:10.1016/S0890-5401(03)00187-1`.

15   Jack H Lutz and Neil Lutz. Algorithmic information, plane Kakeya sets, and conditional dimension. *ACM Transactions on Computation Theory (TOCT)*, 10(2):1–22, 2018.

16   Jack H Lutz and Neil Lutz. Who asked us? How the theory of computing answers questions about analysis. In *Complexity and Approximation*, pages 48–56. Springer, 2020.

17   Jack H Lutz and Elvira Mayordomo. Algorithmic fractal dimensions in geometric measure theory. In *Handbook of Computability and Complexity in Analysis*, pages 271–302. Springer, 2021.

18   Jack H Lutz, Renrui Qi, and Liang Yu. The point-to-set principle and the dimensions of Hamel bases. *Computability*, 13(2):105–112, 2024. `doi:10.3233/COM-210383`.

19   Neil Lutz. Fractal intersections and products via algorithmic dimension. *ACM Transactions on Computation Theory (TOCT)*, 13(3):1–15, 2021.

20   Neil Lutz and Donald M Stull. Projection theorems using effective dimension. In *43rd International Symposium on Mathematical Foundations of Computer Science (MFCS 2018)*, 2018.

21   Neil Lutz and Donald M Stull. Bounding the dimension of points on a line. *Information and Computation*, 275:104601, 2020.

22   Per Martin-Löf. The definition of random sequences. *Information and Control*, 9(6):602–619, 1966.

23   Elvira Mayordomo. A Kolmogorov complexity characterization of constructive Hausdorff dimension. *Information Processing Letters*, 84(1):1–3, 2002.

24   Daniel Osherson and Scott Weinstein. Recognizing strong random reals. *The Review of Symbolic Logic*, 1(1):56–63, 2008.

25   Daniel N Osherson, Michael Stob, and Scott Weinstein. *Systems that Learn: An Introduction to Learning Theory for Cognitive and Computer Scientists*. The MIT Press, 1986.

26   Claus-Peter Schnorr. A unified approach to the definition of random sequences. *Mathematical Systems Theory*, 5(3):246–258, 1971. `doi:10.1007/BF01694181`.

27   Theodore Slaman. Kolmogorov complexity and capacitability of dimension. Report No. 21/2021. Mathematisches Forschungsinstitut Oberwolfach, 2021.

28   Tomasz Steifer. A note on the learning-theoretic characterizations of randomness and convergence. *The Review of Symbolic Logic*, pages 1–16, 2021.

29   Dennis Sullivan. Entropy, Hausdorff measures old and new, and limit sets of geometrically finite Kleinian groups. *Acta Mathematica*, 153(1):259–277, 1984.

30   Claude Tricot. Two definitions of fractional dimension. In *Mathematical Proceedings of the Cambridge Philosophical Society*, volume 91, pages 57–74. Cambridge University Press, 1982.

31   Jean Ville. *Etude critique de la notion de collectif*. Gauthier-Villars Paris, 1939.

32   Francesca Zaffora Blando. A learning-theoretic characterisation of Martin-Löf randomness and Schnorr randomness. *The Review of Symbolic Logic*, 14(2):531–549, 2021.