Algorithmic Identification of Probabilities Is Hard

Laurent Bienvenu[1], Benoît Monin[2], and Alexander Shen[3]

[1] Laboratoire Poncelet
laurent.bienvenu@computability.fr
[2] LIAFA
benoit.monin@liafa.univ-paris-diderot.fr
[3] LIRMM
alexander.shen@lirmm.fr; on leave from IITP RAS

Abstract. Suppose that we are given an infinite binary sequence which is random for a Bernoulli measure of parameter p. By the law of large numbers, the frequency of zeros in the sequence tends to p, and thus we can get better and better approximations of p as we read the sequence. We study in this paper a similar question, but from the viewpoint of inductive inference. We suppose now that p is a computable real, and one asks for more: as we are reading more and more bits of our random sequence, we have to eventually guess the exact parameter p (in the form of its Turing code). Can one do such a thing uniformly for all sequences that are random for computable Bernoulli measures, or even for a 'large enough' fraction of them? In this paper, we give a negative answer to this question. In fact, we prove a very general negative result which extends far beyond the class of Bernoulli measures.

1 Introduction

1.1 Learnability of Sequences

The study of learnability of computable sequences is concerned with the following problem. Suppose we have a black box that generates some infinite computable sequence of bits $X = X(0)X(1)X(2), \dots$ We do not know the program running in the box, and want to guess it looking at finite prefixes

$$X \restriction n = X(0) \dots X(n-1)$$

for increasing n. There could be different programs that produce the same sequence, and it is enough to guess one of them (since there is no way to distinguish between them looking at the output bits). The more bits we see, the more information we have about the sequence. For example, it is hard to say something about a sequence seeing only its first bit 1, but looking at the prefix

$$110010010000111111011010101000$$

P. Auer et al. (Eds.): ALT 2014, LNAI 8776, pp. 85–95, 2014.

one may observe that this is a prefix of the binary expansion of π, and guess that the machine inside the box does exactly that (though the machine may as well produce the binary expansion of, say, 47627751/15160384).

The hope is that, as we gain access to more and more bits, we will *eventually* figure out how the sequence X is generated. More precisely, we hope to have a computable function $\mathfrak{A}$ such that for every computable X, the sequence

$$\mathfrak{A}(X \upharpoonright 1), \mathfrak{A}(X \upharpoonright 2), \mathfrak{A}(X \upharpoonright 3), \ldots$$

converges to a program (=Turing machine) that computes X. This is referred to as *identification in the limit*, and can be understood in two ways:

- Strong success: for every computable X, the above sequence converges to a single program that produces X.
- Weak success: for every computable X, all but finitely many terms of the above sequence are programs that produce X (may be, different ones).

The first type of success is often referred to as *exact* (EX) and the second type as *behaviorally correct* (BC). Either way, such an algorithm $\mathfrak{A}$ does not exist in general. The main obstacle: certain machines are not total (produce only finitely many bits), and distinguishing total machines from non-total ones cannot be done computably. (If we restrict ourselves to some decidable class of total machines, e.g., primitive recursive functions, then exact learning is possible: let $\mathfrak{A}(u)$ be the first machine in the class that is compatible with u.) We refer the reader to [ZZ08] for a detailed survey of learnability of computable functions.

1.2 Learnability of Probability Measures

Recently, Vitanyi and Chater [VC13] proposed to study a related problem. Suppose that instead of a total deterministic machine, the black box contains an *almost total probabilistic machine* M. By "almost total" machine we mean a randomized algorithm that produces an infinite sequence with probability 1. The output distribution of such a machine is a computable probability measure μ_M over the space 2^ω of infinite binary sequences. Again, our ultimate goal is to guess what machine is in the box, i.e., to give a reasonable explanation for the observed sequence X. For example, observing the sequence

$$0001111111100001100000000011111111111$$

one may guess that M is a probabilistic machine that starts with 0 and then chooses each output bit to be equal to the previous one with probability 4/5 (so the change happens with probability 1/5), making all the choices independently.

What should count as a good guess for some observed sequence? Again there is no hope to distinguish between some machine M and another machine M' that has the same output distribution $\mu_{M'} = \mu_M$. So our goal should be to reconstruct the output distribution and not the specific machine.

But even this is too much to ask for. Assume that we have agreed that some machine M is a plausible explanation for some sequence X. Consider another machine M' that starts by tossing a coin and then (depending on the outcome) either generates an infinite sequence of zeros or simulates M'. If X is a plausible output of M, then X is also a plausible output for M', because it may happen (with probability 1/2) that M' simulates M.

A reasonable formalization of 'good guess' is provided by the theory of algorithmic randomness. As Chater and Vitanyi recall, there is a widely accepted formalization of "plausible outputs" for an almost total probabilistic machine with output distribution μ: the notion of Martin-Löf random sequences with respect to μ. These are the sequences which pass all effective statistical tests for the measure μ, also known as μ-Martin-Löf tests. (We assume that the reader is familiar with algorithmic randomness and Kolmogorov complexity. The most useful references for our purposes are [Gác05] and [LV08].) Having this notion in mind, one could look for an algorithm $\mathfrak{A}$ with the following property:

> *for every almost total probabilistic machine M with output distribution μ_M, for μ_M-almost all X, the sequence $\mathfrak{A}(X \restriction 1), \mathfrak{A}(X \restriction 2), \mathfrak{A}(X \restriction 3), \ldots$ identifies in the limit an almost total probabilistic machine M' such that X is $\mu_{M'}$-Martin-Löf random.*

Note that this requirement uses two machines M and M' (more precisely, their output distributions): the first one is used when we speak about "almost all" X, and the second is used in the definition of Martin-Löf randomness. Here M' may differ from M and, moreover, may be different for different X.

Vitanyi and Chater suggest that this can be achieved in the strongest sense (EX): the guesses $\mathfrak{A}(X \restriction n)$ converge to a single code of some machine M'. The main result of this paper says that even a much weaker goal cannot be achieved.

Let us consider a rather weak notion of success: $\mathfrak{A}$ *succeeds* on X if there exists $c > 0$ such that for all sufficiently large n the guess $\mathfrak{A}(X \restriction n)$ is a machine M' such that X is $\mu_{M'}$-Martin-Löf random with randomness deficiency[1] less than c. So the machines $\mathfrak{A}(X \restriction n)$ may be different, we only require that X is Martin-Löf random (with bounded deficiency) for almost all of them. (If almost all machines $\mathfrak{A}(X \restriction n)$ generate the same distribution and X is Martin-Löf random with respect to this distribution, this condition is guaranteed to be true.)

Moreover, we require $\mathfrak{A}$ to be successful only with some positive probability instead of probability 1, and only for machines from some class: for every machine M from this class of machines, $\mathfrak{A}$ is required to succeed with μ_M-probability at least $\delta > 0$, for some δ independent of M.

Of course, this class should not be too narrow: if it contains only one machine M, the algorithm $\mathfrak{A}$ can always produce a code for this machine. The exact conditions on the class will be discussed in the next section.

The proof of this result is quite involved. In the rest of the paper, we specify which classes of machines are considered, present the proof and discuss the consequences of this result.

[1] See below about the version of the randomness deficiency function that we use.

2 Identifying Measures

2.1 Background and Notation

Let us start by providing some notation and background.

We denote by 2^ω the set of infinite binary sequences and by $2^{<\omega}$ the set of finite binary sequences (or *strings*). The length of a string σ is denoted by $|\sigma|$. The n-th element of a sequence $X(0), X(1), \dots$ is $X(n-1)$ (assuming that the length of X is at least n); the string $X \restriction n = X(0)X(1)\dots X(n-1)$ is *n-bit prefix of X*. We write $\sigma \preceq X$ if σ is a prefix of X (of some finite length).

The space 2^ω is endowed with the distance d defined by

$$d(X,Y) = 2^{-\min\{n : X(n) \neq Y(n)\}}$$

This distance is compatible with the product topology generated by *cylinders*

$$[\sigma] = \{X \in 2^\omega \ : \ \sigma \preceq X\}$$

A cylinder is both open and closed (= *clopen*). Thus, any finite union of cylinders is also clopen. It is easy to see, by compactness, that the converse holds: every clopen subset of 2^ω is a finite union of cylinders. We say that a clopen set C has *granularity at most n* if it can be written as a finite union of cylinders $[\sigma]$ with all σ's of length at most n. We denote by Γ_n the family of clopen sets of granularity at most n.

The space of Borel probability measures over 2^ω is denoted by $\mathcal{M}(2^\omega)$. It is equipped with the weak topology. Several classical distances are compatible with this topology; for our purposes, it will be convenient to use the distance ρ, constructed as follows: For $\mu, \nu \in \mathcal{M}(2^\omega)$, let $\rho_n(\mu,\nu)$ (for an integer n) be the quantity

$$\rho_n(\mu,\nu) = \max_{C \in \Gamma_n} |\mu(C) - \nu(C)|$$

and then set

$$\rho(\mu,\nu) = \sum_n 2^{-n} \rho_n(\mu,\nu)$$

The *open* (resp. *closed*) *ball $\mathcal{B}$ of center μ and radius r* is the set of measures ν such that $\rho(\mu,\nu) < r$ (resp. $\rho(\mu,\nu) \leq r$). Note that for any ν in this open (resp. closed) ball, if C is a clopen set of granularity at most n, then $|\mu(C) - \nu(C)| < 2^n\, r$ (resp. $\leq 2^n\, r$). The distance ρ makes $\mathcal{M}(2^\omega)$ a computable compact metric space; its computable points are called *computable probability measures.* A measure is computable if and only if it is the output distribution of some almost total probabilistic Turing machine (see, e.g., [Gác05]). Since $\mathcal{M}(2^\omega)$ is a computable metric space, one can define partial computable functions from some discrete space $\mathcal{X}$ (such as $\mathbb{N}$) to $\mathcal{M}(2^\omega)$ via type-2 computability: a partial function $f :\subseteq \mathcal{X} \to \mathcal{M}(2^\omega)$ is partial computable if there is an algorithm g that for every input $x \in \mathcal{X}$ enumerates a (finite or infinite) list of rational balls[2] $\mathcal{B}_1$, $\mathcal{B}_2$,... in $\mathcal{M}(2^\omega)$ such that $\mathcal{B}_{i+1} \subseteq \mathcal{B}_i$, the radius of $\mathcal{B}_i$ is less than 2^{-i}, and for every x in the domain of f, the list of enumerated balls is infinite and their intersection is the singleton $\{f(x)\}$. (We do not require any specific behavior outside the domain of f.)

Let us introduce two non-standard, but important in this paper, pieces of terminology: having fixed the algorithm g associated to f, we write $\mathrm{err}(f(x)) < \varepsilon$ to mean that the list of balls produced by g on input x contains a ball of radius less than ε (the justification for this notation is that when such a ball is enumerated, should $f(x)$ be defined, we know its value with error at most ε for the distance ρ). When the algorithm g on input x enumerates an empty list of balls, we say that g is *null* on input x.

We denote by K the prefix-free Kolmogorov complexity function. Given a computable measure μ, we call *randomness deficiency of* X *with respect to* μ the quantity

$$\mathbf{d}(X|\mu) = \sup_n \left[\log \frac{1}{\mu([X\restriction n])} - \mathrm{K}(X\restriction n)\right]$$

It is known that $X \in 2^\omega$ is μ-Martin-Löf random (or μ-random for short) if $\mathbf{d}(X|\mu) < \infty$. This definition is slightly non-standard; to get a more standard one, one has to add μ as the condition (with some precautions). However, the above is enough for our purposes.

We say that two measures μ and ν are *orthogonal* if there is a set having μ-measure 1 and ν-measure 0.

If $\mathcal{B}$ is a ball (open or closed) in $\mathcal{M}(2^\omega)$, with center μ and radius r, we define the *estimated deficiency* of X relative to $\mathcal{B}$ by

$$\mathbf{ed}(X|\mathcal{B}) = \sup_n \left[\log \frac{1}{\mu([X\restriction n]) + 2^n\, r} - \mathrm{K}(X\restriction n)\right]$$

Note that $\mathbf{ed}(X|\mathcal{B})$ is a lower bound for $\mathbf{d}(X|\nu)$ for every $\nu \in \mathcal{B}$: we know that the value of $\nu([X\restriction n])$ does not exceed $\mu([X\restriction n]) + 2^n\, r$ for every ν in the ball $\mathcal{B}$. For a fixed pair (X, μ) we have $\lim_{\mathcal{B}\to\mu} \mathbf{ed}(X|\mathcal{B}) = \mathbf{d}(X|\mu)$: if $\mathbf{d}(X|\mu)$ is large, one of the terms (for some n) is large, and the corresponding term in $\mathbf{ed}(X|\mathcal{B})$ is close to it if $\mathcal{B}$ has small radius and contains μ.

Sometimes in the paper we will use the notation $\mathbf{ed}(X|\mathfrak{A}(\sigma))$. By this we mean the supremum of $\mathbf{ed}(X|\mathcal{B})$ over all balls $\mathcal{B}$ output by $\mathfrak{A}$ on input σ.

The next lemma will be useful in the sequel.

Lemma 1 (Randomness deficiency lemma). *Let $\mathcal{B} \subseteq \mathcal{M}(2^\omega)$ be a ball of center μ (rational measure) and rational radius not exceeding r, and let C be a clopen set of granularity at most n. Then for all $X \in C$:*

$$\mathbf{ed}(X|\mathcal{B}) \geq \log \frac{\mu(X\restriction n)}{\mu(X\restriction n) + 2^n r} - \log \mu(C) - K(C, \mu, r, n) - O(1)$$

[2] We fix some natural dense set of finitely representable measures. Rational balls are balls of rational radius with centers in this set. Such balls can also be finitely represented.

Proof. Knowing C, μ, r, n, one can build a prefix-free machine which associates to every string σ of length n such that $[\sigma] \subseteq C$ a a description of size $-\log \mu(X \upharpoonright n) - \log \mu(C)$, so that indeed

$$\sum_{\sigma} 2^{-\log \mu(X \upharpoonright n) - \log \mu(C)} = \frac{1}{\mu(C)} \sum_{\sigma} \mu(\sigma) = 1$$

where the sums are taken over those σ such that $[\sigma] \subseteq C$. This shows that for every such σ of length n, $K(\sigma) \leq -\log \mu(X \upharpoonright n) - \log \mu(C) + K(C, \mu, r, n) - O(1)$. Applying the definition of **ed**, we get, for all $X \in C$

$$\begin{aligned}
\mathbf{ed}(X|\mathcal{B}) &\geq \log \frac{1}{\mu([X \upharpoonright n]) + 2^n\, r} - \mathrm{K}(X \upharpoonright n) \\
&\geq \log \frac{1}{\mu(X \upharpoonright n) + 2^n\, r} + \log \mu(X \upharpoonright n) - \log \mu(C) - K(C, \mu, r, n) - O(1) \\
&\geq \log \frac{\mu(X \upharpoonright n)}{\mu(X \upharpoonright n) + 2^n\, r} - \log \mu(C) - K(C, \mu, r, n) - O(1)
\end{aligned}$$

□

2.2 The Main Theorem

Now we return to the formulation of our main result. The *learning algorithm* is a partial computable function $\mathfrak{A} :\subseteq 2^{<\omega} \to \mathcal{M}(2^\omega)$; it gets the prefix $X \upharpoonright n$ of a sequence X and computes (in type-2 sense) some measure $\mathfrak{A}(X \upharpoonright n)$. (Such a computable function can be converted into an algorithm that, given an input string, produces a program that computes the output measure, and vice versa.) We say that $\mathfrak{A}$ *BC-succeeds* on a sequence $X \in 2^\omega$ if $\mathfrak{A}(X \upharpoonright n)$ outputs the same computable measure μ for all sufficiently large n, and X is Martin-Löf random with respect to μ. This is a weaker requirement that exact (EX) success mentioned above: the algorithm is obliged to produce the same measure (for almost all n), but is not obliged to produce the same machine. Our main result, in its weak form, says that this goal cannot be achieved for all sequences that are random with respect to some computable measure:

Theorem 2. *There is no algorithm $\mathfrak{A}$ that BC-succeeds on every sequence X which is random with respect to some computable measure.*

As we have discussed, we prove a stronger version of this result—stronger in three directions.

First, we require the learning algorithm to succeed only on sequences that are random with respect to measures in some restricted class, for example, the class of Bernoulli measures (the main particular case considered by Chater and Vitanyi).

Second, for each measure μ in this class we do not require the algorithm to succeed on all sequences X that are μ-Martin-Löf random: it is enough that it succeeds with some fixed positive μ-probability (a weaker condition).

Finally, the notion of success on a sequence X is now weaker: we do not require that the algorithm produces (for all sufficiently long inputs) some specific measure, asking only that it gives 'good explanations' for the observed sequence from some point on. More specifically, we say that an algorithm $\mathfrak{A}$ *BD-succeeds* (BD stands for 'bounded deficiency') on some X, if for some c and for all sufficiently large n the measure $\mathfrak{A}(X \upharpoonright n)$ is defined and X is random with deficiency at most c with respect to this measure. Clearly BC-success implies BD-success. (Note that in our definition the randomness deficiency depends only on the measure but not on the algorithm that computes it.)

We now are ready to state our main result in its strong form.

Theorem 3. *Let $\mathcal{M}_0$ be a subspace of $\mathcal{M}(2^\omega)$ with the following properties:*

- *$\mathcal{M}_0$ is effectively closed, i.e., one can enumerate a sequence of open balls in $\mathcal{M}(2^\omega)$ whose union is the complement of $\mathcal{M}_0$.*
- *$\mathcal{M}_0$ is recursively enumerable, i.e., one can enumerate the open balls in $\mathcal{M}(2^\omega)$ which intersect $\mathcal{M}_0$.*
- *every non-empty open subset of $\mathcal{M}_0$ (i.e., a non-empty intersection of an open set in $\mathcal{M}(2^\omega)$ with $\mathcal{M}_0$) contains infinitely many pairwise orthogonal computable measures.*

and let $\delta > 0$. Then there is no algorithm $\mathfrak{A}$ such that for every computable $\mu \in \mathcal{M}_0$, the μ-measure of sequences X on which $\mathfrak{A}$ BD-succeeds is at least δ.

The notion of an recursively enumerable closed set is standard in computable analysis, see [Wei00, Definition 5.1.1].

Note that the hypotheses on the class $\mathcal{M}_0$ are not very restrictive: many standard classes of probability measures have these properties. Bernoulli measures B_p (independent trials with success probability p, where p is a parameter in $[0, 1]$) are an obvious example; so there is no algorithm that can learn all Bernoulli measures (not to speak about all Markov chains). Let us give another interesting example: for every parameter $p \in [0, 1]$, consider measure μ_p associated to the stochastic process which generates a binary sequence bit-by-bit as follows: the first bit is 1, and the conditional probability of 1 after $\sigma 10^k$ is $p/(k+1)$. The class $\mathcal{M}(2^\omega) = \{\mu_p : p \in [0, 1]\}$ satisfies the hypotheses of the theorem.

Note also that these hypotheses are not added for convenience: although they might not be optimal, they cannot be outright removed. If we do not require compactness, then the class of Bernoulli measures B_p with *rational* parameter p would qualify, but it is easy to see that this class admits an algorithm which correctly identifies each of the measures in the class with probability 1. The third condition is important, too. Consider the measures B_0 and B_1 concentrated on the sequences $0000\ldots$ and $1111\ldots$ respectively. Then the class $\mathcal{M}_0 = \{pB_0 + (1-p)B_1 \mid p \in [0, 1]\}$ is indeed effectively compact, but it is obvious that there is an algorithm that succeeds with probability 1 for all measures of that class (in the most strong sense: the first bit determines the entire sequence). For the second condition we do not have a counterexample showing that it is really needed, but it is true for all the natural classes (it is guaranteed to be true if $\mathcal{M}_0$ has a computable dense sequence).

3 The Proof of the Main Theorem

The rest of the paper is devoted to proving Theorem 3. Fix a subset $\mathcal{M}_0$ of $\mathcal{M}(2^\omega)$ satisfying the hypotheses of the theorem, and some $\delta > 0$. In the sequel, by "success" we always mean BD-success.

For every algorithm $\mathfrak{A}$ we consider the set of sequences on which it succeeds. We say that $\mathfrak{A}$ is *δ-good* if this success set has μ-probability at least δ for every $\mu \in \mathcal{M}_0$. We need to show that δ-good algorithms do not exist.

Let us introduce some useful notation. First, let

$$\textsc{Succ}(\mathfrak{A}, c, n) = \big\{X \in 2^\omega \ : \ (X \restriction n) \in \mathrm{dom}(\mathfrak{A}) \wedge \ \mathbf{d}(X|\mathfrak{A}(X \restriction n)) \leq c\big\}$$

be the set of X on which $\mathfrak{A}$ achieves "local success" on the prefix of length n for randomness deficiency c. The success set is then $\bigcup_c \bigcup_N \bigcap_{n \geq N} \textsc{Succ}(\mathfrak{A}, c, n)$.

According to our type-2 definition, the algorithm computing $\mathfrak{A}$ produces (for each input string) a finite or infinite sequence of balls (we assume that i-th ball has radius at most 2^{-i}). We will write '$\mathcal{B} \in \mathfrak{A}(\sigma)$' to signify that on input σ this algorithm enumerates the ball $\mathcal{B}$ at some point. For any function $f : 2^{<\omega} \to [0, 1]$ converging to 0, we define the set $\textsc{Prec}(\mathfrak{A}, f, n)$ of points X which are 'precise enough' in the sense that $\mathfrak{A}(X \restriction n)$ almost outputs a measure:

$$\textsc{Prec}(\mathfrak{A}, f, n) = \{X \in 2^\omega \ : \mathrm{err}(\mathfrak{A}(X \restriction n)) < f(X \restriction n)\}$$

(notice that $\textsc{Prec}(\mathfrak{A}, f, n)$ is a clopen set because the membership of X in $\textsc{Prec}(\mathfrak{A}, f, n)$ is determined fully by the first n bits of X). The specific choice of f (how 'precise' should be the output measure) is discussed later.

In contrast to $\textsc{Prec}$, we define the following "nullity" sets:

$$\textsc{Null}(\mathfrak{A}, N) = \big\{X \in 2^\omega \ : \ \mathfrak{A}(X \restriction n) \text{ is null for every } n \geq N \ \big\}.$$

Proposition 4 (Nullity amplification). *Assume that $\mathfrak{A}$ is a δ-good algorithm, N is an integer, $\eta \geq 0$ is a real number and $\mathcal{B}$ is an open ball intersecting $\mathcal{M}_0$ such that $\mu(\textsc{Null}(\mathfrak{A}, N)) \geq \eta$ for all $\mu \in \mathcal{B} \cap \mathcal{M}_0$. Then there is a non-empty ball $\mathcal{B}' \subseteq \mathcal{B}$ intersecting $\mathcal{M}_0$, an integer $N' \geq N$ and a δ-good algorithm $\mathfrak{A}'$ such that $\mu(\textsc{Null}(\mathfrak{A}', N')) \geq \eta + \delta/2$ for all $\mu \in \mathcal{B}' \cap \mathcal{M}_0$.*

This proposition clearly shows that there can be no δ-good algorithm: if there was one, one could construct by induction (taking for the base case $\eta = 0$, $N = 0$, and $\mathcal{B} =$ any ball intersecting $\mathcal{M}_0$) a sequence of δ-good algorithms $\mathfrak{A}_i$, a non-increasing sequence of balls $\mathcal{B}_i$ intersecting $\mathcal{M}_0$, and a non-decreasing sequence of integers N_i such that $\mu(\textsc{Null}(\mathfrak{A}_i, N_i)) \geq \delta + i \cdot (\delta/2)$ for every $\mu \in \mathcal{B}_i \cap \mathcal{M}_0$, which gives a contradiction for large i. Thus, all we need to do is to prove this proposition.

Proof. Fix $\mathfrak{A}$, N, η and $\mathcal{B}$ as in the hypotheses of the proposition. For $m \geq N$, define a decreasing sequence of effectively open sets $\mathcal{U}_m$ by

$$\mathcal{U}_m = \{\mu \mid (\exists n > m) \big(\mu(\textsc{Prec}(\mathfrak{A}, f, n)) > 1 - \eta - \delta/2\big)\}.$$

The first step of this proof consists in showing that if f is carefully chosen to tend to 0 fast enough, then only finitely many of the $\mathcal{U}_m$ can be dense in $\mathcal{B} \cap \mathcal{M}_0$.

The way we do this is by proving the following fact: if $\mathcal{U}_m$ is dense in $\mathcal{B} \cap \mathcal{M}_0$ for some m, then for every $\mathcal{B}' \subseteq \mathcal{B}$ intersecting $\mathcal{M}_0$, one can effectively find $\mathcal{B}'' \subseteq \mathcal{B}'$ intersecting $\mathcal{M}_0$ such that for all $\mu \in \mathcal{B}''$, $\mu(\textsc{Succ}(\mathfrak{A}, n, n)) < 7\delta/8$ for some $n \geq m \geq N$.

This would yield a contradiction since this would allow us to construct a computable sequence of decreasing balls $\mathcal{B}_m$, all intersecting $\mathcal{M}_0$, where all $\mu \in \mathcal{B}_m$ would be such that $\mu(\textsc{Succ}(\mathfrak{A}, n, n)) < 7\delta/8$ for some $n \geq m$, and thus the intersection of the $\mathcal{B}_m$ would be a computable measure μ^* – belonging to $\mathcal{M}_0$ by closedness of $\mathcal{M}_0$ – for which the success set of $\mathfrak{A}$ has μ^*-measure at most $7\delta/8$, a contradiction.

The definition of f on strings of length n will depend on a "large enough" parameter $s = s(n)$ which we will define later as a computable function of n. Suppose s has already been chosen. We shall first define in terms of s an important auxiliary computable function L. It is computed as follows. For a given n, let $\varepsilon = \min(2^{-n} \cdot \delta/4, r)$ where r is the radius of $\mathcal{B}$.

First, we effectively find $k(\varepsilon)$ rational balls $\mathcal{D}_1, \mathcal{D}_2, \cdots \mathcal{D}_{k(\varepsilon)}$, all intersecting $\mathcal{M}_0$, whose union covers $\mathcal{M}_0$ and for any ball of radius at least ε, one of the $\mathcal{D}_i$ is contained in this ball. (To do this, enumerate all balls with rational center and radius smaller than $\varepsilon/3$. By effective compactness of the space of measures $\mathcal{M}(2^\omega)$ and since $\mathcal{M}_0$ is effectively closed, one can find a finite number of them, call them $\mathcal{D}_1, \mathcal{D}_2, \ldots, \mathcal{D}_{k(\varepsilon)}$, which cover $\mathcal{M}_0$ entirely. Now, let $\mathcal{A}$ be a ball of radius at least ε intersecting $\mathcal{M}_0$ and μ its center. Since μ is at distance $\varepsilon/3$ of some measure $\nu \in \mathcal{M}_0$. But the $\mathcal{D}_i$'s cover $\mathcal{M}_0$, so ν belongs to some ball $\mathcal{D}_i$, and by the triangular inequality, every member of $\mathcal{D}_i$ is at distance at most $2\varepsilon/3$ of μ, hence $\mathcal{D}_i$ is contained in $\mathfrak{A}$).

Then, inside each ball $\mathcal{D}_i$, we effectively find 2^s rational measures $\xi_1^{(i)}, \ldots, \xi_{2^s}^{(i)}$ and pairwise disjoint clopen sets $V_1^{(i)}, \ldots, V_{2^s}^{(i)}$ such that $\xi_j^{(i)}(V_j^{(i)}) > 1 - \delta/8$.

To see that this can be done, observe that the conditions '$\xi_1, \ldots, \xi_s \in \mathcal{B}$', 'the V_i are disjoint', and '$\xi_i(V_i) > 1 - \varepsilon$ for all i' are all Σ^0_1-conditions. Therefore, all we need to argue is that such measures and clopen sets exist. By our assumption on $\mathcal{M}_0$, let $\xi_1, \ldots, \xi_s$ be pairwise orthogonal measures inside $\mathcal{B}$. By definition, this means that for every pair (i, j) with $i \neq j$, there exists a set $S_{i,j} \subseteq 2^\omega$ such that $\xi_i(S_{i,j}) = 1$ and $\xi_j(S_{i,j}) = 0$. For each i, let $S_i = \bigcap_{j \neq i} S_{i,j}$. One can easily check that $\xi_i(S_i) = 1$ for all i and $\xi_i(S_j) = 0$ when $i \neq j$. The measure of a set is the infimum of the measures of open sets covering it. Therefore, for each i there is an open set U_i covering S_i such that $\xi_j(U_i) \leq 2^{-s-1}\varepsilon$ for $i \neq j$ (and of course, $\xi_i(U_i) = 1$ for all i). Now we use the fact that the measure of an open set is the supremum of the measures of the clopen sets it contains. Therefore, for each i there exists a clopen set $U'_i \subseteq U_i$ such that $\xi_i(U'_i) \geq 1 - \varepsilon/2$ (and of course $\xi_i(U'_j) \leq 2^{-s-1}\varepsilon$ for $i \neq j$). Now $V_i = U'_i \setminus \bigcup_{j \neq i} U'_j$ for each i is a clopen set of ξ_i-measure at least $1 - \varepsilon/2 - 2^s \cdot 2^{-s-1}\varepsilon = 1 - \varepsilon$. The pairwise disjointness of the V_i is clear from their definition.

Compute the maximum of the granularities of all the clopen sets $V_j^{(i)}$ for $i \leq k(\varepsilon)$ and $j \leq 2^s$ and denote this maximum by $L(n)$.

Suppose now that for every non-empty $\mathcal{B}' \subseteq \mathcal{B}$ intersecting $\mathcal{M}_0$, there exists some $\mu \in \mathcal{B}'$ and some n,

$$\mu(\textsc{Prec}(\mathfrak{A}, f, n)) > 1 - \eta - \delta/2$$

for some measure $\mu \in \mathcal{B}$ and some $n \geq N$. Set again $\varepsilon = \min(2^{-n} \cdot \delta/4, r)$ and compute a family $\mathcal{D}_1, \mathcal{D}_2, \cdots \mathcal{D}_{k(\varepsilon)}$ intersecting $\mathcal{M}_0$ and whose union covers $\mathcal{M}_0$ so that for any ball $\mathcal{B}$ of radius at least ε, there is some $\mathcal{D}_i \subseteq \mathcal{B}$.

Recall that $\textsc{Prec}(\mathfrak{A}, f, n)$ is a clopen set of granularity n. Thus, if $\rho(\nu, \mu) < 2^{-n} \cdot \delta/4$, then $\nu(\textsc{Prec}(\mathfrak{A}, f, n)) > 1 - \eta - \delta/2 - \delta/4 = 1 - \eta - 3\delta/4$. And thus, by definition of the $\mathcal{D}_i$, there exists i such that for all $\nu \in \mathcal{D}_i$, $\nu(\textsc{Prec}(\mathfrak{A}, f, n)) > 1 - \eta - 3\delta/4$. Moreover, such an i can be found effectively knowing $\textsc{Prec}(\mathfrak{A}, f, n)$ and δ. Fix such an i and set $\mathcal{D} = \mathcal{D}_i$.

Now consider the behaviour of the algorithm $\mathfrak{A}$ on all possible strings σ of length n. On some of these strings, the algorithm does not achieve precision $f(\sigma)$; we ignore such strings. On some others, $\mathfrak{A}(\sigma)$ achieves precision $f(\sigma)$ and thus returns a sequence containing some ball $\mathcal{A}$ of radius less than $f(\sigma)$. Call $\mathcal{A}_1, ..., \mathcal{A}_t$ all such balls (obtained by some $\mathfrak{A}(\sigma)$ with σ of length n). Note that $t \leq 2^n$. Let $\alpha_1, ..., \alpha_t$ be the centers of these balls, and consider their average $\beta = (1/t) \sum_{i \leq t} \alpha_i$. Since the V_i are disjoint and there are 2^s-many of them, by the pigeonhole principle, there exists some j such that $\beta(V_j) \leq 2^{-s}$, and thus $\alpha_i(V_j) \leq t \cdot 2^{-s} \leq 2^{n-s}$ for all i. Fix such a j and set $V = V_j$, and $\xi = \xi_j$.

Recalling that the granularity of V is at most $L(n)$, we can apply the randomness deficiency lemma, we have for all $X \in V$:

$$\begin{aligned}\mathbf{ed}(X|\mathfrak{A}(X\!\upharpoonright\! n)) \geq \log \frac{\alpha_i(X\!\upharpoonright\! L(n))}{\alpha_i(X\!\upharpoonright\! L(n)) + 2^{L(n)} f(X\!\upharpoonright\! n)} - \log \alpha_i(V) \\ -K(V, n, s(n)) - O(1)\end{aligned}$$

where α_i is the center of the ball of radius $f(X \upharpoonright n)$ enumerated by $\mathfrak{A}(X \upharpoonright n)$. And this finally tells us how the function f should be defined: we require that $2^{L(n)} f(X \upharpoonright n)$ is smaller than $\alpha_i(X \upharpoonright L(n))$, so as to make constant the first term of the right-hand-side. It seems to be a circular definition, but it is not the case: we can *define* $f(\sigma)$ to be the first rational q we find such that $\mathfrak{A}(\sigma)$ enumerates a ball of radius at most q and such that the center α of this ball is such that $\alpha(\sigma) > 2^{L(|\sigma|)} q$. This makes f a partial computable function, which is fine for our construction. Note also that $f(\sigma)$ can be undefined if $\mathfrak{A}(\sigma)$ is a measure γ such that $\gamma(\sigma) = 0$, but we need not worry about this case because it automatically makes the algorithm fail on σ (because the γ-deficiency of any extension of σ is infinite).

It remains to evaluate the Kolmogorov complexity of V. What we need to observe that $K(V)$ can be computed from $\textsc{Prec}(\mathfrak{A}, f, n)$, which, being a clopen set of granularity at most n, has complexity at most $2^{n+O(1)}$. Indeed, knowing this set, one can compute the open set of measures ν such that $\nu(\textsc{Prec}(\mathfrak{A}, f, n)) > 1 - \eta - 3\delta/4$ and effectively find a ball $\mathcal{D}$ as above. Then, from $\mathcal{D}$, the sequence of clopen sets $V_1, \ldots, V_{2^s}$ can be effectively computed. Moreover, to choose the V

as above, we need to know β, hence the sequence of measures $\alpha_1, \dots \alpha_t$. But these can also be found knowing $\textsc{Prec}(\mathfrak{A}, f, n)$, by definition of the latter. Thus we have established that $K(V) \leq 2^{n+O(1)}$.

Plugging all these complexity estimates in the above expression, we get

$$\begin{aligned} \mathbf{ed}(X|\mathfrak{A}(X \restriction n)) &\geq s(n) - n - K(s(n)) - O(1) &(1)\\ &\geq s(n) - 2\log(s(n)) - n - O(1) &(2)\end{aligned}$$

Thus, by taking $s(n) = 2n + d$ for some large enough constant d, we get that

$$\mathbf{ed}(X|\mathfrak{A}(X \restriction n)) > n$$

for all $X \in V$. But the clopen set V has ξ-measure at least $1 - \delta/8$, so by definition of the $\mathcal{A}_i$, $\mathfrak{A}$ returns a ξ-inconsistent answer for deficiency level n on a set of ξ-measure at least $1 - \eta - 3\delta/4 - \delta/8$ of strings of length n. Note that this is a Σ^0_1-property of ξ, so we can in fact effectively find a ball $\mathcal{B}''$ intersecting $\mathcal{M}_0$ on which this happens. For every $\nu \in \mathcal{B}''$, $\mathfrak{A}(\sigma)$ is null on a set of strings of ν-measure at least η (by assumption) and is inconsistent on a set of measure at least $1 - \eta - 7\delta/8$, so $\textsc{Succ}(\mathfrak{A}, n, n)$ has a ν-measure of at most $7\delta/8$, which is the contradiction we wanted.

Now, we have reached our first goal which was to show that some $\mathcal{U}_{N'}$ is not dense in $\mathcal{B} \cap \mathcal{M}_0$ for some N'. Note that the $\mathcal{U}_m$ are non-increasing so this further means that there is a ball $\mathcal{B}' \subseteq \mathcal{B}$ such that $\mathcal{B} \cap \mathcal{M}_0$ does not intersect any of the $\mathcal{U}_m$ for $m \geq N'$. By definition, this means that on any measure ν of that ball $\mathcal{B}'$, the algorithm does not reach precision $f(\sigma)$ on a set of strings σ of ν-measure at least $\eta + \delta/2$. Thus, it suffices to consider the algorithm $\mathfrak{A}'$ which on any input σ does the following: it runs $\mathfrak{A}(\sigma)$ until $\mathfrak{A}(\sigma)$ reaches precision $f(\sigma)$. If this never happens, $\mathfrak{A}'(\sigma)$ remains null. If it does, then $\mathfrak{A}'(\sigma)$ returns the same list of balls as $\mathfrak{A}(\sigma)$. Clearly the algorithm $\mathfrak{A}'$ is δ-good since for every σ in the domain of $\mathfrak{A}$, $\mathfrak{A}'(\sigma) = \mathfrak{A}(\sigma)$. But by construction our new algorithm $\mathfrak{A}'$ is such that $\nu(\textsc{Null}(\mathfrak{A}', N')) \geq \eta + \delta/2$ for all $\nu \in \mathcal{B}'$. This finishes the proof. □

Acknowledgements. This publication was made possible through the support of a grant from the John Templeton Foundation. The opinions expressed in this publication are those of the authors and do not necessarily reflect the views of the John Templeton Foundation.

References

[Gác05] Gács, P.: Uniform test of algorithmic randomness over a general space. Theoretical Computer Science 341(1-3), 91–137 (2005)

[LV08] Li, M., Vitányi, P.: An introduction to Kolmogorov complexity and its applications, 3rd edn. Texts in Computer Science. Springer, New York (2008)

[VC13] Vitanyi, P., Chater, N.: Algorithmic identification of probabilities (2013), `http://arxiv.org/abs/1311.7385`

[Wei00] Weihrauch, K.: Computable analysis. Springer, Berlin (2000)

[ZZ08] Zeugmann, T., Zilles, S.: Learning recursive functions: a survey. Theoretical Computer Science 397, 4–56 (2008)