

Two Problems for Sophistication

Peter Bloem, Steven de Rooij and Pieter Adriaans

Abstract

Kolmogorov complexity measures the amount of information in data, but does not distinguish structure from noise. Kolmogorov's definition of the *structure function* was the first attempt to measure only the structural information in data, by measuring the complexity of the smallest model that allows for optimal compression of the data. Since then, many variations of this idea have been proposed, for which we use *sophistication* as an umbrella term. We describe two fundamental problems with existing proposals, showing many of them to be unsound. Consequently, we put forward the view that the problem is fundamental: it may be impossible to objectively quantify the sophistication.

1 Introduction

Kolmogorov complexity gives us a sound definition of the amount of information contained in a binary string. It does not, however, capture what most people would consider complexity. For example, a sequence of a million coin flips will almost certainly have maximal Kolmogorov complexity, even though there is nothing complex about flipping a coin repeatedly. Many scholars have defined additional measures in the spirit of Kolmogorov complexity, aimed at quantifying not *all* information in a binary string, but only the *meaningful*. While this concept has been given many names, we use *sophistication* as an umbrella term. In this paper, we investigate two serious problems with sophistication. We conclude with two arguments suggesting the problems are fundamental, explaining our belief that sophistication cannot be defined in a satisfactory manner.

The Kolmogorov complexity $C(x)$ of a binary string x is, informally, the length of the shortest computer program to print x . This length depends on the choice of programming language, but, by the invariance theorem [15, Section 2.1], only by a constant, independent of x . For sufficiently complex objects, the choice of programming language becomes irrelevant and Kolmogorov complexity becomes an *objective* measure. A definition of sophistication $S(x)$ in the spirit of $C(x)$ should have similar guarantees:

This paper was published at *Algorithmic Learning Theory 2015*. The published version is available from www.springerlink.com.

1. $S(x)$ should count the bits required for an effective description of the structural properties of a binary string.
2. An analogue of invariance should hold: there must be strict limits on how much sophistication can be affected by a change in programming language.
3. There should be no constant c such that $S(x) \leq c$ for every input x . If sophistication is bounded, then knowing its value under one programming language provides no constraints on its value under another language (except it is also bounded).
4. Similarly, there should be no constant c such that $|C(x) - S(x)| \leq c$ for all x , because then sophistication would be equivalent to Kolmogorov complexity.

There have been many proposals for such a measure, all based on a *two-part code*: we encode a *model* in the first part of the code, which is interpreted as a representation of x 's structural properties. The model does not fully specify x , but when combined with the second part of the code, which specifies the noise, the original string becomes fully determined.¹

For any string x , there may be many different two-part codes. The total length can never be less than the Kolmogorov complexity, but it can come close. Figure 1 illustrates the principle. The key to sophistication is to take the representations that come close to the Kolmogorov complexity, the *candidates*, and define the sophistication as the size of the smallest model in this set. However, for most definitions, we can prove that they fail one of the conditions above. For others, we cannot *prove* they conflict with our requirements, but we show these methods only assign substantial sophistication to strings that require an enormous amount of processing to construct.

A valid definition of $S(x)$ must contend with two important issues. First, the details of the way the model is encoded are important. There are two technically distinct approaches; in one of these one has to deal with the so-called “nickname problem” that strangely remains unresolved in several publications. These definitions yield a sophistication that is highly dependent on the chosen programming language, unless special care is taken, as discussed in Section 3.

The second issue is that of striking the right balance between under- and overfitting, which we consider in Section 4. Overfitting is a common problem in statistics, that refers to the tendency to choose a complex model that provides a very good fit to the observed data, but does not generalise well to unseen data. In the case of sophistication, overfitting occurs if the model that determines the sophistication contains much or even all of the noise. In statistics, overfitting is often addressed by penalising complex models. In sophistication, however, such penalties tend to break the balance between structural information and noise, and lead to the opposite problem: underfitting.

¹Some variants deviate from the two-part coding format, see Section 4.3.

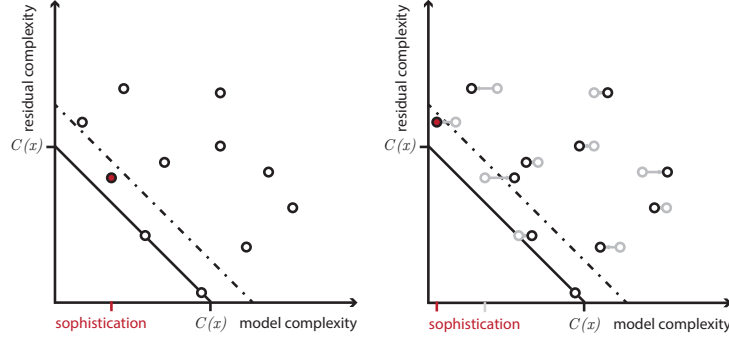


Figure 1: (left) Two-part representations of x by the two components of their code. The Kolmogorov complexity $C(x)$, appearing as a black diagonal, provides a lower bound on the total codelength. We consider only representations that are close to this optimum—called *candidates*—with the threshold represented by a dashed line. The size of the smallest model below the threshold is the sophistication of the data. (right) The same image, after a constant perturbation in the model complexity caused by a change in numbering.

Underfitting occurs when the selected model is simple, but fails to capture all structure in the data. This is also a problem for sophistication because the models under consideration are so powerful. In particular, in any programming language, there are programs that implement an interpreter for *another* language. Such *universal models* are *simple*, since they can be described with a relatively small number of bits, yet are able to represent any data using a code within a constant from the Kolmogorov complexity. Such a two-part representation essentially encodes all information as noise. If complex models are penalized, then the problem becomes to make sure that universal models are not *always* preferred for complex data. The usual workaround is to restrict the set of allowed models, for instance to total functions. While this excludes universal models, it is questionable whether it adequately solves the problem of underfitting in general.

Finally, in Section 5 we argue that while two-part coding can yield useful insights into the structure of the data and identifies some models as poor representations, it is probably not possible to objectively separate structure from noise and identify a *single* model as “best”: many models of different complexities may be reasonable representations. Rather than doggedly trying to “fix” this property of algorithmic statistics, we propose embracing the idea that the data allows for multiple, equivalent interpretations of which information is structured, and which is random, and that there is no such thing as sophistication.

2 Notation

The following notation allows us to generalize across all definitions and variants, save the occasional exception which we will highlight individually.

Let $\mathbb{B} = \{0, 1\}^*$. We deal with partial computable functions $f : \mathbb{B} \times \mathbb{B} \rightarrow \mathbb{B}$, which we also call *models*. f is called *prefix* if $\text{dom}_z(f) = \{y : f(y, z) \neq \infty\}$, is a prefix free set for all z , i.e. no string in $\text{dom}_z(f)$ is a prefix of another. A function f is *total* if $\forall_z \text{dom}_z(f) = \mathbb{B}$. In most cases, we do not use the second argument, and let $f(x) = f(x, \epsilon)$.

A *numbering* is an enumeration of the partial computable functions, denoted by ψ_1, ψ_2, \dots or simply ψ . We fix one canonical numbering ϕ , chosen to be *effective*: i.e. given i and y , we can effectively compute $\phi_i(y)$. We call a numbering ψ *acceptable* if there exist total, computable functions $a, b : \mathbb{N} \rightarrow \mathbb{N}$ with $\forall i : i, \phi_i = \psi_{b(i)}$ and $\psi_i = \phi_{a(i)}$.

A *model class* is a set of indices in a numbering ψ . We define four classes:

- The indices of the partial computable functions $\mathcal{C} = \mathbb{N}$.
- The total functions $\mathcal{T} = \{i : \psi_i \text{ is total}\}$. Note that \mathcal{T} is not computably enumerable.
- \mathcal{K} is an enumerable set such that $\{\psi_i : i \in \mathcal{K}\}$ is the set of all partial computable prefix functions.
- The finite sets: \mathcal{F} is an enumerable set such that $\{\psi_i : i \in \mathcal{F}\}$ is the set of uniform codes for all finite sets.²

Let \bar{x} denote the prefix-encoded representation for x . We require that the mapping satisfies $|\bar{x}| = |x| + O(\log |x|)$ (see eg. [15, Section 1.4]). To simplify notation, we will sometimes conflate natural numbers and binary strings, implicitly using the ordering $(0, \epsilon), (1, 0), (2, 1), (3, 00), (4, 01), \dots$

For technical reasons, we deviate slightly from the traditional notation of Kolmogorov complexity: let \mathcal{M} be a model class and ψ an acceptable numbering, then let $C^{\mathcal{M}, \psi}(x | z) = \min\{|\bar{y}| : \psi_i(y, z) = x, i \in \mathcal{M}\}$, with $C^{\mathcal{M}, \psi}(x) = C^{\mathcal{M}, \psi}(x | \epsilon)$. We omit the numbering when the distinction is not relevant. $C^{\mathcal{C}}(x)$ corresponds to the plain Kolmogorov complexity $C(x)$ and $C^{\mathcal{K}}(x)$ corresponds to the prefix-free version $K(x)$. Note that the notation $C^{(i), \psi}(x)$ represents the smallest two-part description of x using model ψ_i .

We use the principle of a numbering for the purpose normally served by the universal Turing machine. We prefer to work with numberings as it highlights an important issue: while Kolmogorov complexity is invariant to the choice of numbering this property does not immediately carry over to sophistication: for some treatments, the result is highly dependent on the chosen numbering, as we will see in the next section.

3 Inefficient indices

The simplest approach to sophistication would be to ‘open up’ the Kolmogorov complexity and to see which program achieves the smallest description length:

²A uniform code for a set F is a surjective prefix function $f : \{0, 1\}^{\lceil \log |F| \rceil} \rightarrow F$.

the program that *witnesses* the Kolmogorov complexity. This witness is a two-part coding; it consists of a model and an input.

Definition 1 (Index sophistication). Let ψ be an acceptable numbering. Let \mathcal{M} be the model class from which candidates are chosen, and let \mathcal{N} be the model class that determines the minimum achievable complexity. Let c be a fixed constant. The *index sophistication* is:

$$S_{\text{index}}^{\mathcal{M}, \mathcal{N}, \psi, c}(x) = \min \left\{ |i| : C^{\{i\}, \psi}(x) \leq C^{\mathcal{N}, \psi}(x) + c, i \in \mathcal{M} \right\}.$$

When $\mathcal{M} = \mathcal{N}$, we will use $S_{\text{index}}^{\mathcal{M}, \psi, c}$. If the set over which the minimum is taken is empty, the sophistication is undefined.

Koppel and Atlan's treatment [13, 14], where the name *sophistication* originates, follows this basic logic, although it contains idiosyncracies like the use of monotonic models, and an extension to infinite strings. As the subsequent history of sophistication has discarded these, we will not discuss them here.

In [4, 3] Koppel's principle is limited to finite strings, with \mathcal{T} as a model class. The definition is similar to $S_{\text{index}}^{\mathcal{T}, c, \psi, c}$, except the total complexity of a witness (i, y) is measured as $|i| + |y|$ without the cost of delimiting the two. This difference is not relevant to the current discussion. The restriction to \mathcal{T} is a common approach, which avoids underfitting, as discussed in the next section.

Lemma 1. Let S_{index}^{ψ} denote any index sophistication with respect to numbering ψ (with any choice for \mathcal{M} , \mathcal{N} and c). There are acceptable numberings ψ and ξ such that for all x : $|S_{\text{index}}^{\psi}(x) - S_{\text{index}}^{\xi}(x)| \geq \frac{1}{2} \min\{S_{\text{index}}^{\psi}(x), S_{\text{index}}^{\xi}(x)\}$.

Proof. Let $z_i \in \mathbb{B}$ consist of $2^i - 1$ zeroes followed by a one. Define ψ, ξ such that $\psi_j(x) = \phi_i(x)$ for $j = z_{2i}$ and $\xi_j(x) = \phi_i(x)$ for $j = z_{2i+1}$, with all other functions returning ∞ for all inputs. Choose any x and assume w.l.o.g. that $S_{\text{index}}^{\psi}(x) \leq S_{\text{index}}^{\xi}(x)$. By construction, we have $2S_{\text{index}}^{\psi}(x) \leq S_{\text{index}}^{\xi}(x)$. \square

Thus, the length of the index is a very poor indicator of model complexity. For a robust measure, we define the complexity of a function f as in [11, 20] by

$$C^{\mathcal{M}, \psi}(f) = \min\{C^{\mathcal{M}, \psi}(i) : \psi_i = f\}. \quad (1)$$

Lemma 3 in the appendix shows that $C^c(f)$ and $C^{\mathcal{K}}(f)$ are invariant. Note that the perversely inefficient numberings of Lemma 1 are no issue for Kolmogorov complexity: we can use a UTM with a more efficient numbering as a model at only a constant penalty. For sophistication, however, the numbering is crucial.

There are two ways to use $C^{\mathcal{M}}(f)$ for more robust attempts to define sophistication. Confusingly, both are used in the literature. First, we can measure the complexity of the *model* ϕ_i as $C^{\mathcal{K}}(\phi_i)$, which is then the size of the first part of a two-part code describing the data. This approach is used in [7, 8, 20, 9].

Second, we can stick to using the length of the index as the measure of sophistication, but restrict the allowed numberings to those that can represent a given function *efficiently*. This approach is taken by Adriaans in [1], who defines

facticity as $S_{\text{index}}^{\mathcal{C}, \psi, 0}$, but only allows *faithful* numberings. Formally, a faithful numbering has the property that $\forall i \exists j : \psi_i = \psi_j, |j| \leq C^{\mathcal{C}}(\psi_j) + c$, for some constant c . Essentially, this means that a faithful numbering can represent a function f with an index the same length as the Kolmogorov complexity $C^{\mathcal{C}}(f)$.

We prove in the appendix that—contrary to Adriaans’ suggestion—there do exist faithful, acceptable numberings (Lemma 4). However, even choosing a faithful numbering is not enough. The Kolmogorov complexity uses representations of the form $\bar{i}y$, with $\psi_i(y) = x$, where the bar denotes some straightforward prefix encoding to delimit the model description i from its input y . If we define a second prefix encoding \tilde{i} , with $|\bar{i}| - |\tilde{i}|$ unbounded, we can define a second representation $\tilde{u}\tilde{i}y$, with $\psi_u(\tilde{i}y) = \psi_i(y)$, at a constant overhead $|\tilde{u}|$, and gain more than $|\tilde{u}|$ for sufficiently complex strings, resulting again in a bounded sophistication.

We continue with a sophistication that avoids the issues of inefficient indices and of inefficient prefix encodings. We change the definition of index sophistication so that its two-part representations use $C^{\mathcal{K}}(\phi_i)$ bits for the representation of the model. We first introduce the following notation for the \mathcal{M} -Kolmogorov complexity using such compact two-part representations:

$$C_{\text{comp}}^{\mathcal{M}, \psi}(x) = \min\{C^{\mathcal{K}, \psi}(\psi_i) + |y| : \psi_i(y) = x, i \in \mathcal{M}\}.$$

For model classes \mathcal{K} and \mathcal{C} this is equivalent to the existing definition and invariant to the numbering. Note that again, we use $C_{\text{comp}}^{\{i\}, \psi}(x)$ to represent the smallest two-part code using model ψ_i .

Definition 2 (Sophistication).

$$S^{\mathcal{M}, \mathcal{N}, \psi, c}(x) = \min\left\{C^{\mathcal{K}}(\phi_i) : C_{\text{comp}}^{\{i\}, \psi} \leq C_{\text{comp}}^{\mathcal{N}, \psi}(x) + c, i \in \mathcal{M}\right\}.$$

4 Balancing under- and overfitting

In the last section, we began to see the delicate balance between the two code components. We will study this balance, starting with the variant $S^{\mathcal{K}, \psi, c}$, which is not used in the literature, but helps to illustrate the issues we wish to discuss.

\mathcal{K} has optimal representations with all but a constant part of the information in the input and it has optimal representations with all information in the model. The downside to this balance is that it becomes easy to show a lack of invariance. We can tweak the numbering so that models in a specific subset $\mathcal{M}' \subset \mathcal{K}$ become cheaper to represent by an arbitrary amount relative to others: we can ensure that a model in \mathcal{M}' always determines the sophistication. For instance, if we let \mathcal{M}' contain only a universal model we get a bounded sophistication.

Theorem 1 (Underfitting). Let \mathcal{M}, \mathcal{N} be model classes with $\mathcal{M} \subseteq \mathcal{N}$ and let \mathcal{M} contain a universal model ϕ_u , with the property that $\exists c \forall i \in \mathcal{N}, x \in \mathbb{B} : C_{\text{comp}}^{\{u\}, \phi}(x) \leq C_{\text{comp}}^{\mathcal{N}, \phi}(x) + c$. Then, for some numbering ψ , $S^{\mathcal{M}, \mathcal{N}, \psi, c}$ is bounded.

This problem is well known and many treatments avoid it by restricting the model class. Less well known, perhaps, is that the same holds in the other direction: if \mathcal{M}' is the set of *singleton* models—those models that output a single x for an empty input—we get a sophistication equal to the Kolmogorov complexity.

Theorem 2 (Overfitting). Let $\mathcal{X} \subseteq \mathbb{B}$. Let $\mathcal{M} \subseteq \mathcal{N} \subseteq \mathcal{K}$ be model classes where for every $x \in \mathcal{X}$ there is a singleton model $i \in \mathcal{M}$ with $\phi_i(\epsilon) = x$. Then there is a numbering ψ , and a constant c , such that for all $x \in \mathcal{X}$ we have $C^{\mathcal{K}}(x) - S^{\mathcal{M}, \mathcal{N}, \psi, c}(x) \leq c$.

The proofs of both theorems rely on a simple principle: there exist numberings which have the effect of penalizing $C^{\mathcal{K}}(\phi_i)$ for any model outside \mathcal{M}' by an arbitrary constant amount. We can use this to effectively ‘push’ these models outside of the range of candidates, ensuring that, under this numbering, a model in \mathcal{M}' always determines the sophistication. The requirements for \mathcal{M}' are somewhat complex. The following lemma gives a set of sufficient conditions.

Lemma 2. Let \mathcal{M} and \mathcal{N} be any model class, let \mathcal{X} be any set of binary strings and let $D : \mathbb{B} \rightarrow \mathbb{N}$ be a partial computable decoding function with a prefix-free domain that maps function descriptions to their indices in ϕ . Let $\mathcal{M}' = \text{range}(D)$. Further assume there is a constant c such that:

- (a) $\forall m \in \mathcal{M}' : \min\{|p| : \phi_{D(p)} = \phi_m\} \leq C^{\mathcal{K}, \phi}(\phi_m) + c$
- (b) $\forall x \in \mathcal{X} : C_{\text{comp}}^{\mathcal{M}', \phi}(x) - C_{\text{comp}}^{\mathcal{N}, \phi}(x) \leq c$.

Then, there is a ψ such that if $S^{\mathcal{M}, \mathcal{N}, \psi, k}(x)$ is defined, then $S^{\mathcal{M}, \mathcal{N}, \psi, k}(x) = S^{\mathcal{M}', \mathcal{N}, \psi, k}(x)$ up to a constant.

Proof. Pick any $x \in \mathcal{X}$. Let f and g be ϕ -indices such that $f \in \mathcal{M}'$ and $g \notin \mathcal{M}'$ nor is ϕ_g equivalent to any function indexed by \mathcal{M}' . Furthermore let $C_{\text{comp}}^{\{f\}, \phi}(x)$ and $C_{\text{comp}}^{\{g\}, \phi}(x)$ both be within a constant q of $C_{\text{comp}}^{\mathcal{N}, \phi}(x)$. Assumption (b) ensures that \mathcal{M}' always provides such an f .

We will show that for every integer r , there is a numbering ψ such that $C_{\text{comp}}^{\{g'\}, \psi}(x) - C_{\text{comp}}^{\{f'\}, \psi}(x) \geq r$ for all $x \in \mathcal{X}$, where f' and g' are the ψ -indices equivalent to f and g . Thus, for large enough r , ϕ_g is eliminated as a candidate model, while ϕ_f remains in place. Thus, under ψ , a member of \mathcal{M}' determines the sophistication, or the sophistication is undefined.

Let d be a positive constant. We will show later how to choose it to achieve the required result. We define ψ as follows:

$$\psi_0(p) = 0^d 1 D(p), \quad \psi_{0^d 1 i}(p) = \phi_i(p), \quad \psi_j(\cdot) = \infty \text{ if } j \neq 0 \text{ and } j \neq 0^d 1 \dots$$

The key to the proof is the way that the function complexity $C^{\mathcal{K}}(\cdot)$ changes when we change the numbering from ϕ to ψ . For f , the value increases by no more than a fixed constant, but for g , it increases by a constant that we can arbitrarily increase by increasing d .

We will first show that for f , the value does not increase by more than a

constant c_f . Assume w.l.o.g. that $0 \in \mathcal{K}$.

$$\begin{aligned}
C^{\mathcal{K},\psi}(\phi_f) &= \min \{ |\bar{j}q| : \psi_{\psi_j(q)} = \phi_f, j \in \mathcal{K} \} && \text{rewriting (1)} \\
&\leq \min \{ |\bar{0}q| : \psi_{\psi_0(q)} = \phi_f \} && \text{choose } j = 0 \\
&= \min \{ |q| : \phi_{D(q)} = \phi_f \} + |\bar{0}| \\
&\leq C^{\mathcal{K},\phi}(\phi_f) + c_f && \text{by assumption (a).}
\end{aligned}$$

In order to show that for g , we can increase the difference by an arbitrary constant, we first show that, for any z not in the range of ψ_0 , the Kolmogorov complexity itself increases by at least d when we switch from ϕ to ψ :

$$\begin{aligned}
C^{\mathcal{K},\psi}(z) &= \min \{ |\bar{i}y| : \psi_i(y) = z, i \in \mathcal{K} \} && \text{by definition} \\
&= \min \{ |\bar{0}^d 1 j y| : \psi_{0^d 1 j}(y) = z \} && \text{since } z \notin \text{range}(\psi_0) \\
&\geq \min \{ |\bar{j}y| : \phi_j(y) = z \} + d \\
&= C^{\mathcal{K},\phi}(z) + d. && (2)
\end{aligned}$$

We now show the increase in model complexity for g . First, assume $\phi_g \neq \psi_0$:

$$\begin{aligned}
C^{\mathcal{K},\psi}(\phi_g) &= \min \{ C^{\mathcal{K},\psi}(i) : \psi_i = \phi_g \} \\
&= \min \{ C^{\mathcal{K},\psi}(0^d 1 j) : \phi_j = \phi_g \} \\
&= C^{\mathcal{K},\psi}(0^d 1 j) \\
&\geq C^{\mathcal{K},\phi}(0^d 1 j) + d && \text{by (2)} \\
&\geq C^{\mathcal{K},\phi}(j) - c_g + d && \text{since } C^{\mathcal{K}}(j) \leq C^{\mathcal{K}}(0^d 1 j) + c_0 \\
&\geq C^{\mathcal{K},\phi}(\phi_g) - c_g + d.
\end{aligned}$$

Now assume $\phi_g = \psi_0$. We have $C^{\mathcal{K},\psi}(\phi_g) = \min \{ C^{\mathcal{K},\psi}(i) : \psi_i = \psi_0 \} \geq d$. This follows from the fact that the minimum is achieved either at $i = 0$ or at $i = 0^d 1 m$ with $m \notin \mathcal{M}'$. Neither have a representation using a function with a ψ -index without the $0^d 1$ prefix.

Choosing $d \geq r + \max \{ C^{\mathcal{K},\phi}(\psi_0), c_g \} + c_f + 2q$ ensures that for both cases, we have $C^{\mathcal{K},\psi}(\phi_g) \geq C^{\mathcal{K},\phi}(\phi_g) + r + c_f + 2q$. While $C^{\mathcal{K}}(\psi_0)$ depends on the choice of d , we have $C^{\mathcal{K}}(\psi_0) \leq C^{\mathcal{K}}(d) + C^{\mathcal{K}}(D)$, up to a constant, which is in $O(\log d)$, so we can choose d to satisfy the inequality.

Finally, we can show the result:

$$\begin{aligned}
&C_{\text{comp}}^{\{g'\},\psi}(x) - C_{\text{comp}}^{\{f'\},\psi}(x) \\
&= C^{\mathcal{K},\psi}(\phi_g) + \min \{ |y| : \phi_g(y) = x \} - C^{\mathcal{K},\psi}(\phi_f) - \min \{ |y| : \phi_f(y) = x \} \\
&\geq C^{\mathcal{K},\phi}(\phi_g) + r + c_f + 2q + \min \{ |y| : \phi_g(y) = x \} \\
&\quad - C^{\mathcal{K},\phi}(\phi_f) - c_f - \min \{ |y| : \phi_f(y) = x \} \\
&= C_{\text{comp}}^{\{g\},\phi}(x) - C_{\text{comp}}^{\{f\},\phi}(x) + r + 2q \geq r. \quad \square
\end{aligned}$$

Theorems 1 and 2 follow as corollaries. For Theorem 1:

Proof. Let D be a prefix function as in Lemma 2 that returns the index of u for the argument ϵ and ∞ for any other argument. That is, $\mathcal{M}' = \{u\}$. This construction satisfies the conditions 1 and 2 from Lemma 2. Invoking it, we find that there exists an acceptable numbering ψ for which $S^{\mathcal{M}, \mathcal{N}, \psi, k}(x) = S^{\mathcal{M}', \mathcal{N}, \phi, k}(x) + c$. Since \mathcal{M}' contains only a single model, $S^{\mathcal{M}', \mathcal{N}, \phi, c}(x)$ is constant. \square

And for Theorem 2:

Proof. Let x be any string. Given a description of x , we construct some index i such that $\phi_i(\epsilon) = x$ (a singleton for x). Thus, $C^{\mathcal{K}, \psi}(\phi_i) \leq C^{\mathcal{K}, \psi}(x)$ up to a constant. Likewise, given ϕ we can produce x , so that $|C^{\mathcal{K}, \phi}(\phi_i) - C^{\mathcal{K}, \phi}(x)| \leq c$ for some constant c .

We now define a computable function D by $D(\bar{i}y) = j$ where $\phi_j(\epsilon) = \phi_i(y)$ and $i \in \mathcal{K}$, and let \mathcal{M}' be its range. We will show that the two conditions of Lemma 2 hold for the prefix function D .

(a) Let $f \in \mathcal{M}'$ with $\phi_f(\epsilon) = x$. Then $\min\{|p| : \psi_{D(p)} = \phi_f\} = \min\{|\bar{i}q| : \phi_i(q) = x\} = C^{\mathcal{K}}(x) \leq C^{\mathcal{K}}(\phi_f) + c$. (b) On the one hand $C_{\text{comp}}^{\mathcal{M}', \psi}(x) \leq C^{\mathcal{K}}(\phi_f) + |\epsilon| \leq C^{\mathcal{K}}(x) + c$. On the other hand, the witness to $C_{\text{comp}}^{\mathcal{M}, \psi}(x)$ is an effective description of x , so $C^{\mathcal{K}}(x)$ is at most a constant larger.

Now, by Lemma 2 there is a numbering ψ such that we have $S^{\mathcal{M}, \mathcal{N}, \psi, k}(x) = S^{\mathcal{M}', \mathcal{N}, \psi, k}(x) + c$. We observed that $|C^{\mathcal{K}}(\phi_i) - C^{\mathcal{K}}(x)| \leq c_0$ for all singletons, so $S^{\mathcal{M}', \mathcal{N}, \psi, k}(x) \geq C^{\mathcal{K}, \psi}(x) - c_0$. This proves the theorem. \square

Thus, in this balanced sophistication, there is no invariance: all information can be seen as structure, or as noise, depending on the numbering. To avoid these issues, existing proposals upset the balance to exclude or penalize the universal models, and possibly the singleton models.

4.1 Overfitting

We will now review the treatments in the literature that show overfitting. The first is the structure function, proposed by Kolmogorov, most likely the first attempt at separating structure from noise in an objective manner. Kolmogorov defined the following function, using the finite sets \mathcal{F} as models:

$$h_x(\alpha) = \min \{ \log |F| : x \in F, C^{\mathcal{K}}(F) \leq \alpha \}$$

and suggested that the smallest set for which $C^{\mathcal{K}}(F) + \log |F| \leq C^{\mathcal{K}}(x) + c$ holds for some pre-chosen constant c , can be seen as capturing all the structure in x [7]. This is equivalent to the sophistication $S^{\mathcal{F}, \mathcal{K}, \psi, c}(x)$. Theorem 2 shows there are numberings for which this sophistication is always equal to $C^{\mathcal{K}}(x)$. Thus, either this is true for all numberings, or this sophistication is not invariant.

In [8] the structure function is extended to an *algorithmic sufficient statistic*. This is, again, essentially the witness to the sophistication $S^{\mathcal{F}, \mathcal{K}, \psi, c}(x)$. A probabilistic version is also introduced, which uses the model class \mathcal{P} , which indexes the set of functions that compute computable probability semimeasures up to a

multiplicative constant error, yielding $S^{\mathcal{P}, \mathcal{K}, \psi, c}(x)$. For both, Lemma 2 gives us a numbering such that the singleton is always the minimal sufficient statistic.

It may be argued that the slack parameter c in the sophistication, which determines the allowed gap between a candidate representation and the complexity, should depend on the numbering, but this dependence has not been mentioned in the literature and there is no obvious method to choose this constant for a given numbering.

In traditional statistics, overfitting is often addressed by a penalty on complex models. As we have seen, a strong penalty, such as the one imposed by an inefficient prefix encoding of the model, will cause underfitting. A more subtle approach is to allow descriptions that are not self-delimiting. The gap between the smallest self-delimiting description and the smallest non self-delimiting description grows without bound [15, Section 4.5.5], so that some information ends up in the noise, since placing all information in the model results in a self-delimiting, and thus non-optimal description. This eliminates the singletons as viable candidates. This approach is taken by Vítányi [20] and by Adriaans [1]. Such measures reduce the overfitting problem, but they only increase the tendency to underfit. We also pay the price that the models can no longer be equated with probability measures, weakening the link to traditional statistics.

4.2 Underfitting

Universal models are a widely acknowledged problem for sophistication, and most proposals avoid them by limiting the allowed models to exclude them. It is known that there are strings x for which $S^{\mathcal{F}, \mathcal{K}, \psi, c}(x)$, $S^{\mathcal{T}, \psi, c}(x)$ and $S^{\mathcal{T}, \mathcal{K}, \psi, c}(x)$ are close to $|x|$ (up to a logarithmic term). Proofs can be found in [8], [4] and [20] respectively. These are the *absolutely non-stochastic strings* [17]. The existence of these strings is independent of the numbering.

However, the problem of the singletons remains. Only one model class eliminates both the singletons and the universal model: \mathcal{T} . The only proposal we are aware of that uses an efficient model representation *and* excludes the universal models *and* excludes the singletons is: $S^{\mathcal{T}, \mathcal{K}, \psi, c}$, from [20]. While this avoids our proofs of boundedness, there is no evidence that $S^{\mathcal{T}, \mathcal{K}, \psi, c}$ is actually invariant.

While high sophistication strings exist for $S^{\mathcal{T}, \mathcal{K}, \psi, c}$, they may not conform to sophistication's motivating intuition. To show this, we use the concept of depth:

Definition 3 (Depth[5, 2]). Let U be some universal Turing machine, so that $U(\bar{t}y) = \phi_i(y)$. Let U^t be a simulation of this machine, which is allowed to run for at most t steps, and returns 0 if it has not yet finished at that point. Let $C_t^{\mathcal{M}}(x) = \min\{|\bar{t}y| : U^t(\bar{t}y) = x, \phi_i \in \mathcal{M}\}$. The c -depth is $d^{\mathcal{M}, c}(x) = \min\{t : C_t^{\mathcal{M}}(x) - C^{\mathcal{M}}(x) \leq c\}$.

Deep strings are those that can only be optimally compressed with a great investment of time. We note that it is exceedingly unlikely that a deep string is sampled from a shallow distribution [6, 5].

Theorem 3. Let $A(n)$ be the single-argument Ackermann function and c_d some constant. For all k , there is a numbering ψ such that for all strings with depth $d^{\mathcal{C}, c_d}(x) \leq A(C^{\mathcal{C}}(x))$ the sophistication $S^{\mathcal{T}, \mathcal{K}, \psi, k}(x)$ is bounded.

Proof. Let $U(\bar{y})$ be some universal Turing machine, and let $U^A(\bar{y})$ be a simulation of that machine which outputs 0 if the number of steps taken exceeds $A(|\bar{y}|)$. Let u be the index of the function U^A in the standard enumeration.

Let $D(\epsilon) = u$. We can instantiate Lemma 2 with D , $\mathcal{M}' = \{u\}$ and $\mathcal{X} = \{x : d^{\mathcal{C}, c_d}(x) \leq A(C^{\mathcal{C}}(x))\}$. This tells us that there exists a numbering ψ for which $S^{\mathcal{T}, \mathcal{K}, \psi, k}(x) = S^{\mathcal{M}', \mathcal{K}, \psi, k}(x) + |\bar{0}| \leq c$ for all $x \in \mathcal{X}$. \square

This shows that while high-sophistication strings exist, they do not behave as expected. Consider a string that is typical for a shallow model, say some elaborate probabilistic automaton. Under $S^{\mathcal{T}, \mathcal{K}, \psi, c}$, no matter how high the complexity of the automaton, the sophistication is bounded. We could encode the collected works of Shakespeare in its transition graph, and this information would be counted as noise. Any structure simple enough to be exploited within the time bound of the Ackermann function will not be seen as ‘meaningful information’. Only structure so deep that it would take beyond the lifetime of the universe to decompress would count towards sophistication. In the remainder we will refer to strings x with $d^{\mathcal{C}, c}(x) \leq A(C^{\mathcal{C}}(x))$ as *shallow* strings. Note that any string whose shortest program can be run in any time bound represented by a primitive recursive function is shallow.

The relation between $S(x)$ and $d(x)$ is also investigated in [3], where it is shown that within a logarithmic error term on the sophistication and the slack, they are identical. Our point is not the similarity between the two, but that for all practical strings, the sophistication is bounded. This contradicts the intuition that sophistication measures structure, as it seems to suggest that all strings we can possibly hope to understand or generate contain no structure, save a constant amount. The alternative is that under other numberings these strings *do* have structure, but then the sophistication is not invariant.

As for the strings with high sophistication, they have the property that they can be compressed far better with partial functions than with total: they are non-typical for the model class \mathcal{T} . This suggests that the ‘non-stochastic’ property of strings with high sophistication [17, 19] says more about depth and totality than it does about structure and noise.

4.3 Other variants

By moving away from the idea of two-part coding, the mechanics of lemma 2 can be avoided. In [16], the *naive sophistication* is introduced. We will define a generic version, parametrized by model class. Let $C_{\psi_i}(x) = \min \{|y| : \psi_i(y) = x\}$. Then we define the *naive sophistication* as:

$$S_{\text{naive}}^{\mathcal{M}, \psi, c}(x) = \min \{C^{\mathcal{K}, \psi}(\psi_i) : C_{\psi_i}(x) - C^{\mathcal{K}, \psi}(x | i) \leq c, i \in \mathcal{M}\}.$$

The condition now is not that the two-part code length is minimal, but that the *randomness deficiency* $C_{\psi_i}(x) - C^{\mathcal{K}, \psi}(x | i)$ is less than a constant. $S_{\text{naive}}^{\mathcal{T}, \psi, c}(x)$

corresponds to the version in [16]. The switch to the randomness deficiency avoids Theorem 2, but we end up with the same problem as in Theorem 3: for shallow strings $S_{\text{naive}}^{\mathcal{T},\psi,c}$ is defined by the model U^A , and thus bounded.

We cannot show that $S_{\text{naive}}^{\mathcal{T},\psi,c}(x)$ is bounded for shallow strings, but this is only a consequence of the use of sets, not of the switch to randomness deficiency as a condition. Any set sophistication is necessarily lower-bounded by the function $\text{set}(x) = \min \{C^{\mathcal{K}}(F) : x \in F\}$ and if this function were bounded, it would suggest that a finite amount of finite sets contained all strings.

Theorem 4. Let ψ be any acceptable numbering. Then for all shallow x and large enough c , $S_{\text{naive}}^{\mathcal{T},\mathcal{K},\psi,c}(x)$ is bounded and for some constant c_F , we have:

$$\text{set}(x) \leq S_{\text{naive}}^{\mathcal{T},\mathcal{K},\psi,c}(x) \leq C^{\mathcal{K},\psi}(C^{\mathcal{K},\psi}(x)) + c_F.$$

Proof. Let ψ be any acceptable numbering. Let the Turing machine U be defined as $U(\bar{t}y) = \psi_i(y)$ if $i \in \mathcal{K}$, and $U(\bar{t}y) = \infty$ otherwise. Let U^A be derived from U as in section in Section 4.2 and let ϕ_u compute U^A .

For the first part we have $C_{\psi_u}(x) - C^{\mathcal{K},\psi}(x | u) \leq c_0$ for some c_0 , thus for large enough c , $S_{\text{naive}}^{\mathcal{T},\psi,c}(x) \leq C^{\mathcal{K}}(\psi_u)$. For the second part, let $C^{\mathcal{K},\psi}(x) = k$ and $F_k^A = \{x | \exists p : U^A(p) = x, |p| = k\}$. $|F_k^A| \leq |\{p : |p| = k\}|$, so that $\log |F_k^A| \leq k$, which gives us $\log |F_k^A| - C^{\mathcal{K},\psi}(x | F_k^A) \leq c_1$. Thus, for large enough c , $S_{\text{naive}}^{\mathcal{T},\psi,c}(x) \leq C^{\mathcal{K}}(F_k^A)$. From a description of k , we can compute F_k^A with a finite program, so that $C^{\mathcal{K},\psi}(F_k^A) \leq C^{\mathcal{K},\psi}(k) + c_F$, which completes the proof. \square

Note that the constant c only needs to be large enough to ensure that $C^{\mathcal{K},\psi}(x) - C^{\mathcal{K},\psi}(x | u) \leq c$ and $C^{\mathcal{K},\psi}(x) - C^{\mathcal{K},\psi}(x | F_k^A) \leq c$. Since u and F_k^A are generally of no value in computing x , c is likely very small.

Another approach is the *coarse sophistication* [4], defined in [16] as:

$$S_{\text{coarse}}^{\mathcal{M},\mathcal{N},\psi}(x) = \min_c \{S^{\mathcal{M},\mathcal{N},\psi,c}(x) + c\}.$$

Again, this variant avoids the pitfalls of Theorem 2. If there are candidates that are as good as the singletons but with smaller size by more than a constant, the constant penalty c will eventually be much less than the gain for the simpler witness, and the singletons will not determine the coarse sophistication. The coarse sophistication is within a logarithmic term of the busy beaver depth [4]. As with the naive sophistication, we can show that for shallow strings, the total function version is bounded, and the set version grows very slowly:

Theorem 5. Let ψ be any acceptable numbering. For all shallow x , $S_{\text{coarse}}^{\mathcal{T},\mathcal{K},\psi}(x)$ is bounded and there is a constant c_F such that:

$$\text{set}(x) \leq s_{\text{coarse}}^{\mathcal{T},\mathcal{K},\psi}(x) \leq 2C^{\mathcal{K},\psi}(C^{\mathcal{K},\psi}(x)) + c_F.$$

Proof. Let ψ be any acceptable numbering and define U^A , ϕ_u and F_k^A as in the proof of Theorem 4. For the first part, we know that for some constant c_0 , $C_{\text{comp}}^{\{u\},\psi}(x) \leq C_{\text{comp}}^{\mathcal{K},\psi}(x) + c_0$ so that $S^{\mathcal{T},\mathcal{K},\psi,c_0}(x) \leq C^{\mathcal{K},\psi}(\phi_u)$, thus $S_{\text{coarse}}^{\mathcal{T},\mathcal{K},\psi}(x) \leq$

$C^{\mathcal{K},\psi}(\phi_u) + c_0$. For the second part, we know that for some c_1 , $C^{\mathcal{K},\psi}(F_k^A) + \log|F_k^A| \leq C^{\mathcal{K},\psi}(k) + C^{\mathcal{K},\psi}(x) + c_1$, so that $S^{\mathcal{F},\mathcal{K},\psi,C^{\mathcal{K}}(k)+c_1}(x) \leq C^{\mathcal{K},\psi}(F_k^A)$. Thus, $S_{\text{coarse}}^{\mathcal{F},\mathcal{K},\psi}(x) \leq C^{\mathcal{K},\psi}(F_k^A) + C^{\mathcal{K},\psi}(k) + c_1 = 2C^{\mathcal{K},\psi}(C^{\mathcal{K},\psi}(x)) + c_F$. \square

In [18], Vereshchagin proposes a *strongly algorithmic sufficient statistic*. Where the regular algorithmic sufficient statistic F from [8] has $C^{\mathcal{K}}(F | x)$ constant, the strong variant imposes the stronger requirement that $C^{\mathcal{T}}(F | x)$ is also constant. This reduces the problems of underfitting discussed in this section, but since $C^{\mathcal{T}}(\{x\} | x)$ is bounded, by Theorem 2, overfitting remains a problem: there exist numberings under which the singletons are the only candidates.

Finally, *effective complexity* [9], proposed by Gell-Man and Lloyd, was formulated from the perspective of physics, but fits the mold of sophistication. The model class consists of all computable probability distributions on finite sets. The complexity of the model is measured by its Kolmogorov complexity, avoiding the problems of Section 3. Theorem 2, however, still applies to effective complexity. Unlike other sophistication measures, it is not the candidate with the smallest model which is chosen, but the one which reproduces the data within the shortest time. Thus, if there are multiple candidates, this approach would likely favor the singletons. In [10], the authors abandon this approach, and note that the choice from the set of candidates is a subjective one, which depends on context, which is in line with the view we express in the next section.

5 Discussion and conclusion

We have criticized existing measures of sophistication and shown technical problems with all of them. But that does not in itself mean that it should be impossible to come up with a sound measure. The common intuition, starting with the structure function, appears to be that the crucial property is whether a string is typical for a model, and that this typicality can be tested: another random choice from that model should select a string with the same structure. This idea is bold, but not unreasonable. Nevertheless, we offer the opinion that such a clean-cut separation *cannot* be made to work. We provide two arguments.

For the first argument, we take a generative perspective. We can generate data from a model $\phi_i, i \in \mathcal{K}$, by feeding it random bits until it produces an output. We will call the resulting probability distribution p_i . Call a sophistication *consistent* if, for sufficiently large data, it reflects the complexity of the source of the data. Now, let $\phi_u(\bar{y}) = \phi_i(y)$ and sample from p_u . Then the initial bits will determine the prefix encoded index \bar{i} of the function $\phi_{\bar{i}}$ that ϕ_u will subsequently emulate, and the remaining bits are used as inputs to $\phi_{\bar{i}}$. We now ask, what should be the sophistication of the resulting data?

Certainly, if we have to judge based only on the data, we cannot exclude the possibility that the data was sampled from p_u : after all, it was. Yet, neither can we deny that it may have come from p_i , as again, it did! Eliminating the universal models does not solve this problem: the same argument holds if ϕ_u

indexes, for instance, only those models computable by finite automata. Any model that dominates a set of other models creates this kind of ambiguity.

Consider the following metaphor. We are given a a bitmap image of the painting *Impression of a Sunrise*. There are many good models for this string, from very generic to very specific. Sophistication suggests that we can choose one of these as the objective, intrinsic model of the data. The universal model says that it is ‘some compressible, finite object’. Another might say that it is ‘an image’. Even more specific would be ‘a painting’, ‘a Monet’, or specifically ‘the painting *Impression of a Sunrise*’. A sound sophistication should be able to select one of these as the proper representation of structure in the data, and disqualify the others as over- or underfitting. But how should we be able to say that the data is intrinsically more of a painting than an image? More of a Monet than a painting? Intuitively, such distinctions require further assumptions, or a second sample from the same distribution.

The second reason we doubt sophistication is more technical. Consider the set of all possible two-part representations of x . When the numbering is changed, the codelength of the model part of all these representations will change. This is illustrated in the second diagram in Figure 1. The invariance theorem expresses that this change is limited by a constant term. However, even this small shift can push some representations out of the acceptable region (indicated by the dashed line), and pull others in. This may lead to a different representation determining the sophistication, one whose *total* codelength is close to what it was before, but whose *model* codelength can be anywhere between 0 and $C^c(x)$. If such jumps can occur, the sophistication is not invariant. And while we cannot *prove* in general that such jumps can always occur, there seems to be no reason to believe that they do not. Indeed, in [3] it is shown that logarithmic changes in the slack parameter can already cause these effects.

So we take a skeptical view of sophistication. Note that part of the theory is fine: there is nothing wrong with evaluating models for the data by comparing their two-part code lengths. In fact, the randomness deficiency $-\log p_i(x) - C^{\mathcal{K}}(x \mid i)$ has a direct statistical interpretation as a measure of counterevidence—under p_i , the probability of a randomness deficiency above k is less than 2^{-k} [6, Lemma 6]. In the Monet example above, this will allow us to disqualify the model expressing that the data is actually, say, a recording of jazz music.

But fundamental problems arise as soon as a hard cut-off is introduced on how far we are allowed to deviate from the minimum determined by the Kolmogorov complexity. In our opinion, a lot of measures taken in the literature, such as restricting the model class or introducing model penalties, complicate the method and make problems harder to analyse, without actually addressing the fundamental issue. This is dangerous: if such ad-hoc fixes result in a theory that is hard to prove either wrong or right, it creates an artificial dead end for a valuable area of research. When the hard cut-off on candidates is avoided, however, all such measures are no longer necessary. What remains is an elegant theory that can be used to sift through all possible models, disproving most while retaining a select number of interesting candidates for our further

consideration.

Acknowledgements

This publication was supported by the Dutch national program COMMIT and by the Netherlands eScience center. We thank Tom Sterkenburg for interesting discussions.

References

- [1] P. Adriaans. Facticity as the amount of self-descriptive information in a data set, 2012. arXiv:1203.2245.
- [2] L. Antunes, L. Fortnow, D. van Melkebeek, and N.V. Vinodchandran. Computational depth: concept and applications. *Th. Comp. Sc.*, 354(3):391–404, 2006.
- [3] L.F.C. Antunes, B. Bauwens, A. Souto, and A. Teixeira. Sophistication vs logical depth, 2013. <http://arxiv.org/abs/1304.8046>.
- [4] L.F.C. Antunes and L. Fortnow. Sophistication revisited. *Theory Comput. Syst.*, 45(1):150–161, 2009.
- [5] C.H. Bennett. Logical depth and physical complexity. In *The Universal Turing Machine: A Half-Century Survey*. Oxford University Press, 1988.
- [6] P. Bloem, F. Mota, S. de Rooij, L. Antunes, and P. Adriaans. A safe approximation for Kolmogorov complexity. In *ALT*, pages 336–350, 2014.
- [7] T.M. Cover. Kolmogorov complexity, data compression, and inference. In *The Impact of Processing Techniques on Communications*, pages 23–33. Springer, 1985.
- [8] P. Gács, J. Tromp, and P.M.B. Vitányi. Algorithmic statistics. *IEEE Tr. Inf. Th.*, 47(6):2443–2463, 2001.
- [9] M. Gell-Mann and S. Lloyd. Information measures, effective complexity, and total information. *Complexity*, 2(1):44–52, 1996.
- [10] M. Gell-Mann and S. Lloyd. Effective complexity. *Nonextensive Entropy-Interdisciplinary Applications*, by Edited by Murray Gell-Mann and C Tsallis, pp. 440. Oxford University Press, Apr 2004. ISBN-10: 0195159764. ISBN-13: 9780195159769, 1, 2004.
- [11] P. Grünwald and P.M.B. Vitányi. Shannon information and Kolmogorov complexity, 2004. arXiv:cs/0410002.
- [12] S.C. Kleene. On notation for ordinal numbers. *J. Symb. Log.*, pages 150–155, 1938.

- [13] M. Koppel. Structure. In *The Universal Turing Machine: A Half-Century Survey*. Oxford University Press, 1988.
- [14] M. Koppel and H. Atlan. An almost machine-independent theory of program-length complexity, sophistication, and induction. *Inf. Sci.*, 56(1-3):23–33, 1991.
- [15] M. Li and P.M.B. Vitányi. *An introduction to Kolmogorov complexity and its applications*. Springer-Verlag, 1993.
- [16] F. Mota, S. Aaronson, L.F.C. Antunes, and A. Souto. Sophistication as randomness deficiency. In *DCFS 2013*, pages 172–181, 2013.
- [17] A. Kh. Shen. The concept of (α, β) -stochasticity in the Kolmogorov sense, and its properties. *Soviet Math. Dokl.*, 28(1):295–299, 1983.
- [18] Nikolay Vereshchagin. Algorithmic minimal sufficient statistics: a new approach. *Theory of Computing Systems*, pages 1–19, 2015.
- [19] N.K. Vereshchagin and P.M.B. Vitányi. Kolmogorov’s structure functions and model selection. *IEEE Tr. Inf. Th.*, 50(12):3265–3290, 2004.
- [20] P.M.B. Vitányi. Meaningful information. *IEEE Tr. Inf. Th.*, 52(10), 2004.

A Appendix

Lemma 3 (Invariance of function complexity). Let ψ and η be any two acceptable numberings. Let f be any partial computable function. There exists a constant c independent of f such that

$$|C^{\mathcal{K}, \psi}(f) - C^{\mathcal{K}, \eta}(f)| \leq c \text{ and } |C^{e, \psi}(f) - C^{e, \eta}(f)| \leq c.$$

Proof. Let $g(i)$ be the function such that $\psi_i = \eta_{g(i)}$.

$$\begin{aligned} C^{e, \psi}(f) &= \min \{C^{e, \psi}(i) : \psi_i = f\} \geq \min \{C^{e, \eta}(i) : \psi_i = f\} - c \\ &= \min \{C^{e, \eta}(i) : \eta_{g(i)} = f\} - c = \min \{C^{e, \eta}(g(i)) : \eta_{g(i)} = f\} - c' \\ &\geq \min \{C^{e, \eta}(j) : \eta_j = f\} - c' = C^{e, \eta}(f). \end{aligned}$$

Reverse ψ and η for the opposite inequality. The same proof holds for $C^{\mathcal{K}}$. \square

Lemma 4. There are faithful acceptable numberings.

Proof. Let $d \in \mathbb{N}$ be an index such that $\phi_d(y) = \infty$ for all y . Define

$$\psi_q = \begin{cases} \phi_{\phi_i(p)} & \text{if } q \text{ can be written as } \bar{i}p \text{ and } \phi_i(p) < \infty, \\ \phi_d & \text{otherwise.} \end{cases}$$

It may seem that the second line requires a test whether $\phi_i(p)$ halts, for ψ to be acceptable, but as we will show below, this is not the case.

To show that ψ is faithful, pick any function f . Then

$$\begin{aligned} C^{\mathcal{C}, \Phi}(f) &= \min\{C^{\mathcal{C}, \Phi}(i) : \phi_i = f\} = \min\{\min\{|\bar{a}b| : \phi_a(b) = i\} : \phi_i = f\} \\ &= \min\{|\bar{a}b| : \phi_{\phi_a(b)} = f\} = \min\{|\bar{a}b| : \psi_{\bar{a}b} = f\}. \end{aligned}$$

This shows there is a sufficiently small ψ index.

To show that ψ is acceptable, let $\phi_j(z) = z$. Then a ϕ -index i can be mapped to a ψ -index with $r(i) = \bar{j}i$, so that $\psi_{r(i)}(y) = \psi_{\bar{j}i}(y) = \phi_i(y)$. For the reverse, define $\phi_v(\bar{i}p, y) = \phi_{\phi_i(p)}(y)$. For fixed $\bar{i}p$, the s_m^n -theorem [12] states that we can compute the h such that $\phi_h(y) = \phi_v(\bar{i}p, y)$. Let $h(\bar{i}p)$ denote this index as a function of the program; further define $h(q) = d$ if q cannot be expressed as $\bar{i}p$. By construction h is total and computable. To check that the mapping returns the correct function, rewrite $\phi_{h(\bar{i}p)}(y) = \phi_v(\bar{i}p, y) = \phi_{\phi_i(p)}(y) = \psi_{\bar{i}p}(y)$. Note that if q can be written as $\bar{i}p$, but $\phi_i(p)$ diverges, $h(q)$ will still return a function, but one which doesn't halt, making it equivalent to ϕ_d as required. \square