Electronic Colloquium on Computational Complexity, Report No. 133 (2014)

Mutual Dimension*

Adam Case and Jack H. Lutz
Department of Computer Science
Iowa State University
Ames, IA 50011 USA

Abstract

We define the lower and upper *mutual dimensions* $mdim(x : y)$ and $Mdim(x : y)$ between any two points x and y in Euclidean space. Intuitively these are the lower and upper densities of the algorithmic information shared by x and y. We show that these quantities satisfy the main desiderata for a satisfactory measure of mutual algorithmic information. Our main theorem, the *data processing inequality* for mutual dimension, says that, if $f : \mathbb{R}^m \to \mathbb{R}^n$ is computable and Lipschitz, then the inequalities $mdim(f(x) : y) \leq mdim(x : y)$ and $Mdim(f(x) : y) \leq Mdim(x : y)$ hold for all $x \in \mathbb{R}^m$ and $y \in \mathbb{R}^t$. We use this inequality and related inequalities that we prove in like fashion to establish conditions under which various classes of computable functions on Euclidean space preserve or otherwise transform mutual dimensions between points.

1 Introduction

Recent interactions among computability theory, algorithmic information theory, and geometric measure theory have assigned a *dimension* $dim(x)$ and a *strong dimension* $Dim(x)$ to each individual point x in a Euclidean space $\mathbb{R}^n$. These dimensions, which are real numbers satisfying $0 \leq dim(x) \leq Dim(x) \leq n$, have been shown to be geometrically meaningful. For example, the *classical* Hausdorff dimension $dim_H(E)$ of any set $E \subseteq \mathbb{R}^n$ that is a union of Π^0_1 (computably closed) sets is now known [18, 12] to admit the pointwise characterization

$$dim_H(E) = \sup_{x \in E} dim(x).$$

More recent investigations of the dimensions of individual points in Euclidean space have shed light on connectivity [20, 24], self-similar fractals [19, 6], rectifiability of curves [10, 22, 9], and Brownian motion [13].

In their original formulations [18, 1], $dim(x)$ is $cdim(\{x\})$ and $Dim(x)$ is $cDim(\{x\})$, where $cdim$ and $cDim$ are constructive versions of classical Hausdorff and packing dimensions [7], respectively. Accordingly, $dim(x)$ and $Dim(x)$ are also called *constructive fractal dimensions*. It is often most convenient to think of these dimensions in terms of the Kolmogorov complexity characterization

*This research was supported in part by National Science Foundation Grants 0652569, 1143830, and 124705. Part of the second author's work was done during a sabbatical at Caltech and the Isaac Newton Institute for Mathematical Sciences at the University of Cambridge. A preliminary version of part of this material was reported at the 2013 Symposium on Theoretical Aspects of Computer Science in Kiel, Germany.

ISSN 1433-8092

theorems

$$dim(x) = \liminf_{r\to\infty} \frac{K_r(x)}{r},\ Dim(x) = \limsup_{r\to\infty} \frac{K_r(x)}{r}, \tag{1.1}$$

where $K_r(x)$, the Kolmogorov complexity of x at precision r, is defined later in this introduction [21, 1, 19]. These characterizations support the intuition that $dim(x)$ and $Dim(x)$ are the lower and upper *densities of algorithmic information* in the point x.

In this paper we move the pointwise theory of dimension forward in two ways. We formulate and investigate the *mutual dimensions* — intuitively, the lower and upper densities of shared algorithmic information — between two points in Euclidean space, and we investigate the *conservation* of dimensions and mutual dimensions by computable functions on Euclidean space. We expect this to contribute to both computable analysis — the theory of scientific computing [3] — and algorithmic information theory.

The analyses of many computational scenarios call for quantitative measures of the degree to which two objects are correlated. In classical (Shannon) information theory, the most useful such measure is the *mutual information* $I(X : Y)$ between two probability spaces X and Y [5]. In the algorithmic information theory of finite strings, the (*algorithmic*) *mutual information* $I(x : y)$ between two *individual* strings $x, y \in \{0,1\}^*$ plays an analogous role [17]. Under modest assumptions, if x and y are drawn from probability spaces X and Y of strings respectively, then the expected value of $I(x : y)$ is very close to $I(X : Y)$ [17]. In this sense algorithmic mutual information is a refinement of Shannon mutual information.

Our formulation of mutual dimensions in Euclidean space is based on the algorithmic mutual information $I(x : y)$, but we do not use the seemingly obvious approach of using the binary expansions of the real coordinates of points in Euclidean space. It has been known since Turing's famous correction [25] that binary notation is not a suitable representation for the arguments and values of computable functions on the reals. (See also [14, 26].) This is why the characterization theorems (1.1) use $K_r(x)$, the *Kolmogorov complexity* of a point $x \in \mathbb{R}^n$ at precision r, which is the minimum Kolmogorov complexity $K(q)$ — defined in a standard way [17] using a standard binary string representation of q — for all rational points $q \in \mathbb{Q}^n \cap B_{2^{-r}}(x)$, where $B_{2^{-r}}(x)$ is the open ball of radius 2^{-r} about x. For the same reason we base our development here on the *mutual information* $I_r(x : y)$ between points $x \in \mathbb{R}^m$ and $y \in \mathbb{R}^n$ at *precision* r. This is the minimum value of the algorithmic mutual information $I(p : q)$ for all rational points $p \in \mathbb{Q}^m \cap B_{2^{-r}}(x)$ and $q \in \mathbb{Q}^n \cap B_{2^{-r}}(y)$. Intuitively, while there are infinitely many pairs of rational points in these balls and many of these pairs will contain a great deal of "spurious" mutual information (e.g., *any* finite message can be encoded into both elements of such a pair), a pair of rational points p and q achieving the minimum $I(p : q) = I_r(x : y)$ will only share information that their proximities to x and y force them to share. Sections 3 and 4 below develop the ideas that we have sketched in this paragraph, along with some elements of the fine-scale geometry of algorithmic information in Euclidean space that are needed for our results. A modest generalization of Levin's coding theorem (Theorem 3.1 below) is essential for this work.

In analogy with the characterizations (1.1) we define our mutual dimensions as the lower and upper densities of algorithmic mutual information,

$$mdim(x : y) = \liminf_{r\to\infty} \frac{I_r(x : y)}{r},\ Mdim(x : y) = \limsup_{r\to\infty} \frac{I_r(x : y)}{r}, \tag{1.2}$$

in section 5. We also prove in that section that these quantities satisfy all but one of the desiderata (e.g., see [2]) for any satisfactory notion of mutual information.

We save the most important desideratum — our main theorem — for section 6. This is the data processing inequality for mutual dimension (actually two inequalities, one for $mdim$ and one for $Mdim$). The data processing inequality of Shannon information theory [5] says that, for any two probability spaces X and Y and any function $f : X \to Y$,

$$I(f(X) : Y) \leq I(X : Y) \tag{1.3}$$

i.e., the induced probability space $f(X)$ obtained by "processing the information in X through f" does not share any more information with Y than X shares with Y. The data processing inequality of algorithmic information theory [17] says that, for any computable partial function $f : \{0,1\}^* \to \{0,1\}^*$, there is a constant $c_f \in \mathbb{N}$ (essentially the number of bits in a program that computes f) such that, for all strings $x \in dom\, f$ and $y \in \{0,1\}^*$,

$$I(f(x) : y) \leq I(x : y) + c_f. \tag{1.4}$$

That is, modulo the constant c_f, $f(x)$ contains no more information about y than x contains about y.

The data processing inequality for points in Euclidean space is a theorem about functions $f : \mathbb{R}^m \to \mathbb{R}^n$ that are computable in the sense of computable analysis [3, 14, 26]. Briefly, an *oracle* for a point $x \in \mathbb{R}^m$ is a function $g_x : \mathbb{N} \to \mathbb{Q}^m$ such that $|g_x(r) - x| \leq 2^{-r}$ holds for all $r \in \mathbb{N}$. A function $f : \mathbb{R}^m \to \mathbb{R}^n$ is *computable* if there is an oracle Turing machine M such that, for every $x \in \mathbb{R}^m$ and every oracle g_x for x, the function $r \mapsto M^{g_x}(r)$ is an oracle for $f(x)$.

Given (1.2), (1.3), and (1.4), it is natural to conjecture that, for every computable function $f : \mathbb{R}^m \to \mathbb{R}^n$, the inequalities

$$mdim(f(x) : y) \leq mdim(x : y), \quad Mdim(f(x) : y) \leq Mdim(x : y) \tag{1.5}$$

hold for all $x \in \mathbb{R}^m$ and $y \in \mathbb{R}^t$. However, this is not the case. For a simple example, there exist computable functions $f : \mathbb{R} \to \mathbb{R}^2$ that are *space-filling*, e.g., satisfy $[0,1]^2 \subseteq range\, f$ [4]. For such a function f we can choose $x \in \mathbb{R}$ such that $dim(f(x)) = 2$. Letting $y = f(x)$, we then have

$$mdim(f(x) : y) = dim(f(x)) = 2 > 1 \geq Dim(x) \geq Mdim(x : y),$$

whence both inequalities in (1.5) fail.

The difficulty here is that the above function f is extremely sensitive to its input, and this enables it to compress a great deal of "sparse" high-precision information about its input x into "dense" lower-precision information about its output $f(x)$. Many theorems of mathematical analysis exclude such excessively sensitive functions by assuming a given function f to be *Lipschitz*, meaning that there is a real number $c > 0$ such that, for all x and x', $|f(x) - f(x')| \leq c|x - x'|$. This turns out to be exactly what is needed here. In section 6 we prove prove the *data processing inequality* for mutual dimension (Theorem 6.1), which says that the conditions (1.5) hold for every function $f : \mathbb{R}^m \to \mathbb{R}^n$ that is computable and Lipschitz. In fact, we derive the data processing inequality from the more general *modulus processing lemma* (Lemma 6.4). This lemma yields quantitative variants of the data processing inequality for other classes of functions. For example, we use the modulus processing lemma to prove that, if $f : \mathbb{R}^m \to \mathbb{R}^n$ is *Hölder with exponent* α (meaning that $0 < \alpha \leq 1$ and there is a real number $c > 0$ such that $|f(x) - f(x')| \leq c|x - x'|^\alpha$ for all $x, x' \in \mathbb{R}^m$), then the inequalities

$$mdim(f(x):y) \leq \frac{1}{\alpha} mdim(x:y), \quad Mdim(f(x):y) \leq \frac{1}{\alpha} Mdim(x:y) \tag{1.6}$$

hold for all $x \in \mathbb{R}^m$ and $y \in \mathbb{R}^t$.

In section 7 we derive *reverse* data processing inequalities, e.g., giving conditions under which $mdim(x:y) \leq mdim(f(x):y)$. In section 8 we use data processing inequalities and their reverses to explore conditions under which computable functions on Euclidean space preserve, approximately preserve, or otherwise transform mutual dimensions between points.

2 Preliminaries

We write $\mathbb{Z}$ for the set of integers, $\mathbb{N}$ for the set of non-negative integers, $\mathbb{Q}$ for the set of rationals, $\mathbb{R}$ for the set of reals, and $\mathbb{R}^n$ for the set of all n-vectors $(x_1, x_2, \cdots, x_n)$ such that each $x_i \in \mathbb{R}$. Our logarithms are in base 2. We denote the cardinality of a set A, the length of a string $s \in \{0,1\}^*$, and the distance between two points $x, y \in \mathbb{R}^n$ (using the Euclidean metric) by $|A|$, $|s|$, and $|x - y|$ respectively. We also denote the i^{th} string in $\{0,1\}^*$ by s_i.

Our use of Turing machines is strictly limited to self-delimiting (or prefix) machines. Because of this, we refer to a self-delimiting Turing machine simply as a Turing machine. We refer the reader to Li and Vitanyi [17] for a detailed explanation of how self-delimiting Turing machines work.

The (*conditional*) *Kolmogorov complexity* of a string $x \in \{0,1\}^*$ *given* a string $y \in \{0,1\}^*$ with respect to a Turing machine M is

$$K_M(x \,|\, y) = \min\{|\pi| \,\Big|\, \pi \in \{0,1\}^* \text{ and } M(\pi, y) = x\}.$$

The *Kolmogorov complexity* of x with respect to M is $K_M(x) = K_M(x \,|\, \lambda)$, where λ is the *empty string.* A Turing machine M' is *optimal* if, for every Turing machine M, there is a constant $c_M \in \mathbb{N}$ such that, for all $x \in \{0,1\}^*$,

$$K_{M'}(x) \leq K_M(x) + c_M.$$

We call c_M an *optimality constant* for M. It is well-known that every universal Turing machine is optimal [17]. Following standard practice, we fix a universal, hence optimal, Turing machine U; we omit it from the notation, writing $K(x) = K_U(x)$ and $K(x \,|\, y) = K_U(x \,|\, y)$; and we call these the *Kolmogorov complexity* of x and the (*conditional*) *Kolmogorov complexity* of x *given* y, respectively.

The *joint Kolmogorov complexity* of two strings $x, y \in \{0,1\}^*$ is

$$K(x,y) = K(\langle x, y \rangle),$$

where $\langle \cdot, \cdot \rangle$ is some standard pairing function for encoding two strings. Gács [8] proved the useful identity

$$K(x,y) = K(x) + K(y \,|\, \langle x, K(x) \rangle) + O(1). \tag{2.1}$$

The *universal a priori probability* of a set $S \subseteq \{0,1\}^*$ is

$$\mathbf{m}(S) = \sum_{U(\pi) \in S} 2^{-|\pi|}.$$

Since we are using self-delimiting machines, the Kraft inequality tells us that $\mathbf{m}(\{0,1\}^*) \leq 1$. The *universal a priori probability* of a string $x \in \{0,1\}^*$ is $\mathbf{m}(\mathrm{x}) = \mathbf{m}(\{\mathrm{x}\})$. For $r \in \mathbb{N}$, we write $K(r)$ for $K(s_r)$ and $\mathbf{m}(r)$ for $\mathbf{m}(s_r)$. It is well known that there is a constant $c_0 \in \mathbb{N}$ such that $K(x) \leq |x| + 2\log(1+|x|) + c_0$, and hence $K(r) \leq \log(1+r) + 2\log(1+\log(1+r)) + c_0$, hold for all $x \in \{0,1\}^*$ and $r \in \mathbb{N}$.

Levin's coding lemma plays an important role in section 3.

Lemma 2.1 (coding lemma [15, 16])**.** *If $A \subseteq \{0,1\}^* \times \mathbb{N}$ is computably enumerable and satisfies $\Sigma_{(x,l)\in A} 2^{-l} \leq 1$, then there is a Turing machine M such that, for each $(x,l) \in A$, there is a string $\pi \in \{0,1\}^l$ satisfying $M(\pi) = x$.*

3 Kolmogorov Complexity in Euclidean Space

We begin by developing some elements of the fine-scale geometry of algorithmic information in Euclidean space. In this context it is convenient to regard the Kolmogorov complexity of a set of strings to be the number of bits required to specify *some* element of the set.

Definition (Shen and Vereshchagin [23]). The *Kolmogorov complexity* of a set $S \subseteq \{0,1\}^*$ is

$$K(S) = \min\{K(x) \,|\, x \in S\}.$$

Note that $S \subseteq T$ implies $K(S) \geq K(T)$. Intuitively, small sets may require "higher resolution" than large sets.

We need a generalization of Levin's coding theorem [15, 16] that is applicable to certain systems of disjoint sets.

Notation. Let $B \subseteq \mathbb{N} \times \mathbb{N} \times \{0,1\}^*$ and $r, s \in \mathbb{N}$.

1. The (r,t)-*block* of B is the set $B_{r,t} = \{x \in \{0,1\}^* \,|\, (r,t,x) \in B\}$.
2. The r^{th} *layer* of B is the sequence $B_r = (B_{r,t} \,|\, t \in \mathbb{N})$.

Definition. A *layered disjoint system* (LDS) is a set $B \subseteq \mathbb{N}\times\mathbb{N}\times\{0,1\}^*$ such that, for all $r, s, t \in \mathbb{N}$,

$$s \neq t \Rightarrow B_{r,s} \cap B_{r,t} = \emptyset.$$

Note that this definition only requires the sets within each layer of B to be disjoint.

Theorem 3.1 (LDS coding theorem)**.** *For every computably enumerable layered disjoint system B there is a constant $c_B \in \mathbb{N}$ such that, for all $r, t \in \mathbb{N}$,*

$$K(B_{r,t}) \leq \log \frac{1}{\mathbf{m}(B_{r,t})} + K(r) + c_B.$$

Proof. Assume the hypothesis, and fix a computable enumeration of B. For each $r, t \in \mathbb{N}$ such that $B_{r,t} \neq \emptyset$, let $x_{r,t}$ be the first element of $B_{r,t}$ to appear in this enumeration. Let A be the set of all ordered pairs $(x_{r,t}, j+k+2)$ such that $r, t, j, k \in \mathbb{N}$, $B_{r,t} \neq \emptyset$, $k \geq K(r)$, and $\mathbf{m}(B_{r,t}) \geq 2^{-j}$. It is clear that A is computably enumerable.

For each $r, t \in \mathbb{N}$, let

$$j_{r,t} = \min\{j \in \mathbb{N} \,\big|\, \mathbf{m}(B_{r,t}) > 2^{-j}\},$$

noting that $j_{r,t} = \infty$ if $B_{r,t} = \emptyset$. For all $r, t \in \mathbb{N}$ such that $B_{r,t} \neq \emptyset$, we have

$$\begin{aligned}\sum_{\substack{l \in \mathbb{N} \\ (x_{r,t}, l) \in A}} 2^{-l} &= \sum_{j=j_{r,t}}^{\infty} \sum_{k=K(r)}^{\infty} 2^{-(j+k+2)} \\ &= \sum_{k=K(r)}^{\infty} 2^{-(k+1)} \sum_{j=j_{r,t}}^{\infty} 2^{-(j+1)} \\ &= 2^{-K(r)} 2^{-j_{r,t}} \\ &< 2^{-K(r)} \mathbf{m}(B_{r,t}).\end{aligned}$$

Since the sets in each layer B_r of B are disjoint, it follows that

$$\begin{aligned}\sum_{(x,l) \in A} 2^{-l} &\leq \sum_{r=0}^{\infty} \sum_{t=0}^{\infty} 2^{-K(r)} \mathbf{m}(B_{r,t}) \\ &= \sum_{r=0} 2^{-K(r)} \sum_{t=0}^{\infty} \mathbf{m}(B_{r,t}) \\ &= \sum_{r=0}^{\infty} 2^{-K(r)} \mathbf{m}\left(\bigcup_{t=0}^{\infty} B_{r,t}\right) \\ &\leq \sum_{r=0}^{\infty} 2^{-K(r)} \mathbf{m}(\{0,1\}^*) \\ &\leq \sum_{r=0}^{\infty} 2^{-K(r)} \\ &\leq \sum_{r=0}^{\infty} \mathbf{m}(r) \\ &= \mathbf{m}(\{0,1\}^*) \\ &\leq 1.\end{aligned}$$

We have now shown that the set A satisfies the hypothesis of Lemma 2.1. Let M be a Turing machine for A as in that lemma, and let $c_B = c_M + 3$, where c_M is an optimality constant for M. To see that c_B affirms the theorem, let $r, t \in \mathbb{N}$ be such that $B_{r,t} \neq \emptyset$. (The theorem is trivial if $B_{r,t} = \emptyset$, since the right-hand side is infinite.) Then $(x_{r,t}, j_{r,t} + K(r) + 2) \in A$, so there is a program

$\pi \in \{0,1\}^{j_{r,t}+K(r)+2}$ such that $M(\pi) = x_{r,t}$. We thus have

$$\begin{aligned}
K(B_{r,t}) &\leq K(x_{r,t}) \\
&\leq K_M(x_{r,t}) + c_M \\
&\leq |\pi| + c_M \\
&= j_{r,t} + K(r) + 2 + c_M \\
&= \lfloor \log \frac{1}{\mathbf{m}}(B_{r,t}) \rfloor + 1 + K(r) + 2 + c_M \\
&\leq \log \frac{1}{\mathbf{m}(B_{r,t})} + K(r) + c_B. \qquad \square
\end{aligned}$$

Note that Levin's coding theorem [15, 16], the nontrivial part of which says that $K(x) \leq \log \frac{1}{\mathbf{m}(x)} + O(1)$, is the special case $B_{r,t} = \{s_t\}$ of the LDS coding theorem.

Our next objective is to use the LDS coding theorem to obtain useful bounds on the number of times that the value $K(S)$ is attained or approximated.

Definition. Let $S \subseteq \{0,1\}^*$ and $d \in \mathbb{N}$.

1. A *d-approximate K-minimizer* of S is a string $x \in S$ for which $K(x) \leq K(S) + d$.
2. A *K-minimizer* of S is a 0-approximate K-minimizer of S.

We use the LDS coding theorem to prove the following.

Theorem 3.2. *For every computably enumerable layered disjoint system B there is a constant $c_B \in \mathbb{N}$ such that, for all $r, t, d \in \mathbb{N}$, the block $B_{r,t}$ has at most $2^{d+K(r)+c_B}$ d-approximate K-minimizers.*

Proof. Let B be a computably enumerable LDS, and let c_B be as in the LDS coding theorem. Let $r, t, d \in \mathbb{N}$, and let N be the number of d-approximate K-minimizers of the block $B_{r,t}$. Then

$$\mathbf{m}(B_{r,t}) \geq N \cdot 2^{-(K(B_{r,t})+d)},$$

so the LDS coding theorem tells us that

$$\begin{aligned}
K(B_{r,t}) &\leq \log \frac{1}{N \cdot 2^{-(K(B_{r,t})+d)}} + K(r) + c_B \\
&= K(B_{r,t}) + d - \log N + K(r) + c_B.
\end{aligned}$$

This implies that

$$\log N \leq d + K(r) + c_B,$$

whence

$$N \leq 2^{d+K(r)+c_B}. \qquad \square$$

We now lift our terminology and notation to Euclidean space $\mathbb{R}^n$. In this context, a layered disjoint system is a set $B \subseteq \mathbb{N} \times \mathbb{N} \times \mathbb{R}^n$ such that, for all $r, s, t \in \mathbb{N}$,

$$s \neq t \Rightarrow B_{r,s} \cap B_{r,t} = \emptyset.$$

We lift our Kolmogorov complexity notation and terminology to $\mathbb{R}^n$ in two steps:

1. Lifting to $\mathbb{Q}^n$: Each rational point $q \in \mathbb{Q}^n$ is encoded as a string $x \in \{0,1\}^*$ in a natural way. We then write $K(q)$ for $K(x)$. In this manner, $K(S)$, $\mathbf{m}(S)$, K-minimizers, and d-approximate K-minimizers are all defined for sets $S \subseteq \mathbb{Q}^n$.

2. Lifting to $\mathbb{R}^n$. For $S \subseteq \mathbb{R}^n$, we define $K(S) = K(S \cap \mathbb{Q}^n)$ and $\mathbf{m}(S) = \mathbf{m}(S \cap \mathbb{Q}^n)$. Similarly, a K-minimizer for S is a K-minimizer for $S \cap \mathbb{Q}^n$, etc.

For each $r \in \mathbb{N}$ and each $m = (m_1, \ldots, m_n) \in \mathbb{Z}^n$, let

$$Q_m^{(r)} = [m_1 \cdot 2^{-r},\, (m_1+1) \cdot 2^{-r}) \times \cdots \times [m_n \cdot 2^{-r},\, (m_n+1) \cdot 2^{-r})$$

be the *r-dyadic cube* at m. Note that each $Q_m^{(r)}$ is "half-open, half-closed" in such a way that, for each $r \in \mathbb{N}$, the family

$$\mathcal{Q}^{(r)} = \{Q_m^{(r)} \mid m \in \mathbb{Z}^n\}$$

is a partition of $\mathbb{R}^n$. It follows that (modulo trivial encoding) the collection

$$\mathcal{Q} = \{Q_m^{(r)} \mid r \in \mathbb{N} \text{ and } m \in \mathbb{Z}^n\}$$

of all *dyadic cubes* is a layered disjoint system whose rth layer is $\mathcal{Q}^{(r)}$. Moreover, the set

$$\{(r, m, q) \in \mathbb{N} \times \mathbb{Z}^n \times \mathbb{Q}^n \mid q \in Q_m^{(r)}\}$$

is decidable, so Theorem 3.2 has the following useful consequence.

Corollary 3.3. *There is a constant $c \in \mathbb{N}$ such that, for all $r, d \in \mathbb{N}$, no r-dyadic cube has more than $2^{d+K(r)+c}$ d-approximate K-minimizers. In particular, no r-dyadic cube has more than $2^{K(r)+c}$ K-minimizers.*

The Kolmogorov complexity of an arbitrary point in Euclidean space depends on both the point and a precision parameter.

Definition. Let $x \in \mathbb{R}^n$ and $r \in \mathbb{N}$. The *Kolmogorov complexity* of x at *precision* r is

$$K_r(x) = K(B_{2^{-r}}(x)).$$

That is, $K_r(x)$ is the number of bits required to specify *some* rational point in the open ball $B_{2^{-r}}(x)$. Note that, for each $q \in \mathbb{Q}^n$, $K_r(q) \nearrow K(q)$ as $r \to \infty$.

Given an open ball B of radius ρ and a real number $\alpha > 0$, we write αB for the ball with the same center as B and radius $\alpha\rho$. We also write $\overline{B}$ for the topological closure of B.

The definition of $K_r(x)$ directs our attention to the Kolmgorov complexities of arbitrary balls of radius 2^{-r} in Euclidean space. The following easy fact is repeatedly useful in this context.

Observation 3.4. *For every open ball $B \subseteq \mathbb{R}^n$ of radius 2^{-r},*

$$B \cap 2^{-(r+\lceil \frac{1}{2}\log n \rceil)}\mathbb{Z}^n \neq \emptyset.$$

Proof. If B is such a ball, then the expanded ball

$$B' = 2^{r+\lceil \frac{1}{2}\log n \rceil} B$$

has radius

$$2^{\lceil \frac{1}{2}\log n\rceil} > 2^{\frac{1}{2}\log n - 1} = \frac{\sqrt{n}}{2}.$$

This implies that

$$B' \cap \mathbb{Z}^n \neq \emptyset,$$

whence

$$\begin{aligned} B \cap 2^{-(r+\lceil \frac{1}{2}\log n\rceil)}\mathbb{Z}^n &= 2^{-(r+\lceil \frac{1}{2}\log n\rceil)}(B' \cap \mathbb{Z}^n) \\ &\neq \emptyset. \end{aligned}$$

□

We use Observation 3.4 to establish the following connection between the complexities of cubes and the complexities of balls.

Lemma 3.5. *There is a constant $c \in \mathbb{N}$ such that, for every $r \in \mathbb{N}$, every r-dyadic cube Q, and every open ball $B \subseteq \mathbb{R}^n$ of radius 2^{-r} that intersects Q,*

$$K(B) \leq K(Q) + K(r) + c.$$

Proof. Fix a computable enumeration $m_0, m_1, m_2, \cdots$ of $\mathbb{Z}^n$ satisfying $|m_i| \leq |m_{i+1}|$ for all $i \in \mathbb{N}$. Note that, for all $i \in \mathbb{N}$,

$$i < |\overline{B_{|m_i|}(0)} \cap \mathbb{Z}^n| \leq (2|m_i| + 1)^n. \tag{3.1}$$

Let $l = \lceil \frac{1}{2}\log n\rceil$, and let M be a self-delimiting Turing machine such that, if $U(\pi_1) = q \in \mathbb{Q}^n$ and $U(\pi_2) = r \in \mathbb{N}$, then, for all $i \in \mathbb{N}$,

$$M(\pi_1\pi_2 0^{|s_i|}1 s_i) = q + 2^{-(r+l)}m_i. \tag{3.2}$$

Let $c = 2\lceil 2n\log(1+\sqrt{n})\rceil + 1 + c_M$, where c_M is an optimality constant for M.

Now assume the hypothesis, and let q be a K-minimizer of Q. Observation 3.4 tells us that there is a point $m \in \mathbb{Z}^n$ such that $2^{-(r+l)}m \in B - q$. Then $|2^{-(r+l)}m|$ is the distance from a point in B to the point $q \in \mathbb{Q}$, so

$$|m| = 2^{r+l}|2^{-(r+l)}m| \leq 2^{r+l} diam(B \cup Q).$$

Since $B \cap Q \neq \emptyset$, it follows that

$$\begin{aligned} |m| &\leq 2^{r+l}[diam(B) + diam(Q)] \\ &= 2^l(2+\sqrt{n}) \\ &\leq \frac{\sqrt{n}}{2}(2+\sqrt{n}) \\ &= \frac{n}{2} + \sqrt{n}. \end{aligned} \tag{3.3}$$

It is crucial here that this bound does not depend on B, Q, or r.

Choose $i \in \mathbb{N}$ such that $m_i = m$. By (3.1) and (3.3),

$$i < (2(\frac{n}{2} + \sqrt{n}) + 1)^n = (1+\sqrt{n})^{2n}. \tag{3.4}$$

Now let $\pi = \pi_1\pi_2 0^{|s_i|}1s_i$, where π_1 and π_2 are minimum-length programs for q and r, respectively. By (3.2) we have

$$M(\pi) = q + 2^{-(r+l)}m_i \in B.$$

It follows by (3.4) that

$$\begin{aligned} K(B) &\leq K(q + 2^{-(r+l)}m_i) \\ &\leq K_M(q + 2^{-(r+l)}m_i) + c_M \\ &\leq |\pi| + c_M \\ &= K(q) + K(r) + 2|s_i| + 1 + c_M \\ &= K(Q) + K(r) + 2\lceil 2n \log(1 + \sqrt{n})\rceil + 1 + c_M \\ &= K(Q) + K(r) + c. \end{aligned}$$

□

Theorem 3.6. *There is a constant $c \in \mathbb{N}$ such that, for all $r, d \in \mathbb{N}$, no open ball of radius 2^{-r} has more than $2^{d+2K(r)+c}$ d-approximate K-minimizers. In particular, no open ball of radius 2^{-r} has more than $2^{2K(r)+c}$ K-minimizers.*

Proof. Let B be an open ball of radius 2^{-r}, let Q be a r-dyadic cube such that $B \cap Q = \emptyset$, and let $u = K(B) - K(Q)$. There are at most $2^{d+u+K(r)+c'}$ $(d+u)$-approximate K-minimizers $q \in \mathbb{Q}$ of Q such that $K(q) \leq K(Q) + d + u = K(B) + d$ where $c' \in \mathbb{N}$ is a constant from Corollary 3.3. Therefore, there are at most $2^{d+u+K(r)+c'}$ d-approximate K-minimizers of B in $Q \cap B$.

Observe that it takes at most $3^n = 2^{n \log 3}$ r-dyadic cubes to cover B. By Lemma 3.5, $u \leq K(r) + c''$, where $c'' \in \mathbb{N}$ is a constant. Therefore, it follows that B has at most $2^{d+2K(r)+c}$ d-approximate K-minimizers where $c = c' + c'' + n \log 3$. In particular, B has at most $2^{2K(r)+c}$ K-minimizers. □

Lemma 3.5 also gives a slightly simplified proof of the known upper bound on $K_r(x)$.

Observation 3.7 ([19])**.** *For all $x \in \mathbb{R}^n$, $K_r(x) \leq nr + o(r)$.*

Proof. Let c be a constant of Lemma 3.5, let $x = (x_1, \ldots, x_n) \in \mathbb{R}^n$, and let

$$\gamma_x = \max\{|x_i| + 1 \,\big|\, 1 \leq i \leq n\}.$$

For each $r \in \mathbb{N}$, let $m(r) = (m_1, \ldots, m_n)$ be the unique $m \in \mathbb{Z}^n$ such that $x \in \mathbb{Q}_m^{(r)}$. Then, for each $r \in \mathbb{N}$ and $1 \leq i \leq n$, we have $|m_i| \leq 2^r\gamma_x$. It follows easily from this that there is a constant $c' \in \mathbb{N}$ such that, for every $r \in \mathbb{N}$,

$$K(m(r)) \leq n(\log(2^r\gamma_x) + 2\log\log(2^r\gamma_x)) + c_1. \tag{3.5}$$

There is clearly a constant $c_2 \in \mathbb{N}$ such that, for every $r \in \mathbb{N}$,

$$K(2^{-r}m(r)) \leq K(m(r)) + K(r) + c_2. \tag{3.6}$$

By (3.5), (3.6), and Lemma 3.5 we now have

$$\begin{aligned}K_r(x) &= K(B_{2^{-r}}(x)) \\ &\leq K(Q^{(r)}_{m(r)}) + K(r) + c \\ &\leq K(m(r)) + K(r) + c \\ &\leq nr + \epsilon(r),\end{aligned}$$

where

$$\begin{aligned}\epsilon(r) &= n(\log \gamma_x + 2\log\log(2^r\gamma_x)) + 2K(r) + c + c_1 + c_2. \\ &= o(r)\end{aligned}$$

as $r \to \infty$. □

Lemma 3.8. *There is a constant $c \in \mathbb{N}$ such that, for all $r, s \in \mathbb{N}$, $x \in \mathbb{R}^n$, and $q \in B_{2^{-r}}(x)$,*

$$K_{r+s}(x) \leq K(q) + ns + K(r) + a_s,$$

where $a_s = K(s) + 2\log(\lceil \frac{1}{2}\log n\rceil + s + 3) + n(\lceil \frac{1}{2}\log n\rceil + 3) + K(n) + 2\log n + c$.

Proof. Fix a computable enumeration $m_0, m_1, m_2, \cdots$ of $\mathbb{Z}^n$ satisfying $|m_i| \leq |m_{i+1}|$ for all $i \in \mathbb{N}$. Note that, for all $i \in \mathbb{N}$,

$$i < |\overline{B_{|m_i|}(0)} \cap \mathbb{Z}^n| \leq (2|m_i| + 1)^n. \tag{3.7}$$

Let $l = \lceil \frac{1}{2}\log n\rceil$, and let M be a self-delimiting Turing machine such that, if $U(\pi_1) = q \in \mathbb{Q}^n$, $U(\pi_2) = r \in \mathbb{N}$, $U(\pi_3) = s \in \mathbb{N}$, $U(\pi_4) = n \in \mathbb{N}$, and $U(\pi_5) = i \in \mathbb{N}$, then

$$M(\pi_1\pi_2\pi_3\pi_4\pi_5) = q + 2^{-(r+s+l+1)}m_i. \tag{3.8}$$

Let $a_s = 2n(\lceil \frac{1}{2}\log n\rceil + s + 3) + 1 + c_M$, where c_M is an optimality constant for M.

Now assume the hypothesis. Observation 3.4 tells us that there is a point $m \in \mathbb{Z}^n$ such that $2^{-(r+s+l)}m \in B_{2^{-(r+s)}}(x) - q$. Then $|2^{-(r+s+l)}m|$ is the distance from a point in $B_{2^{-(r+s)}}(x)$ to the point q, so

$$\begin{aligned}|m| &= 2^{r+s+l}|2^{-(r+s+l)}m| \\ &\leq 2^{r+s+l}(2^{-r} + 2^{-(r+s)}) \\ &= 2^{s+l}(1 + 2^{-s}) \\ &= 2^l(2^s + 1) \\ &\leq 2^l 2^{s+1} \\ &\leq 2^{l+s+1}.\end{aligned} \tag{3.9}$$

Choose $i \in \mathbb{N}$ such that $m_i = m$. By (3.7) and (3.9),

$$i < (2|m_i| + 1)^n \leq (2(2^{l+s+1}) + 1)^n = (2^{l+s+2} + 1)^n. \tag{3.10}$$

Now let $\pi = \pi_1\pi_2\pi_3\pi_4\pi_5$, where π_1, π_2, π_3, π_4, and π_5 are minimum-length programs for q, r,

s, n, and i, respectively. By (3.8) we have

$$M(\pi) = q + 2^{-(r+s+l+1)} m_i \in B_{2^{-(r+s)}}(x). \tag{3.11}$$

Therefore, (3.11) and optimality tell us that

$$\begin{aligned} K_{r+s}(x) &= K(B_{2^{-(r+s)}}(x)) \\ &\le K(q + 2^{-(r+l)} m_i) \\ &\le K_M(q + 2^{-(r+l)} m_i) + c_M \\ &= |\pi| + c_M \\ &= K(q) + K(r) + K(s) + K(n) + K(i) + c_M. \end{aligned}$$

As noted in section 2, there is a constant $c_0 \in \mathbb{N}$ such that

$$K(i) \le \log(1 + i) + 2\log(1 + \log(1 + i)) + c_0.$$

It follows by (3.10) that

$$\begin{aligned} K(i) &\le n\log(1 + 2^{l+s+2}) + 2\log(1 + n\log(1 + 2^{l+s+2})) + c_0 \\ &\le n(l + s + 3) + 2\log(1 + n(l + s + 3)) + c_0 \\ &\le n(l + s + 3) + 2(1 + \log n + \log(l + s + 3)) + c_0 \\ &= ns + n(l + 3) + 2\log n + 2\log(l + s + 3) + c_0 + 2. \end{aligned}$$

Letting $c = c_M + c_0 + 2$, it follows that

$$K_{r+s}(x) \le K(q) + ns + a_s,$$

where $a_s = K(s) + 2\log(l + s + 3) + n(l + 3) + K(n) + 2\log n + c$. □

The following corollary says roughly that, in $\mathbb{R}^n$, precision can be improved by ns bits by adding ns bits of specification.

Corollary 3.9. *There is a constant $c \in \mathbb{N}$ such that, for all $r, s \in \mathbb{N}$ and $x \in \mathbb{R}^n$,*

$$K_{r+s}(x) \le K_r(x) + ns + b_s,$$

where $b_s = a_s + K(r)$ and a_s is as in Lemma 3.8.

4 Algorithmic Mutual Information in Euclidean Space

This section develops the algorithmic mutual information between points in Euclidean space at a given precision. As in section 3, we assume that rational points $q \in \mathbb{Q}^n$ are encoded as binary strings in some natural way. Mutual information between rational points is then defined from conditional Kolmogorov complexity in the standard way [17] as follows.

Definition. Let $p \in \mathbb{Q}^m$, $r \in \mathbb{Q}^n$, $s \in \mathbb{Q}^t$.

1. The *mutual information* between p and q is

$$I(p : q) = K(q) - K(q \mid p).$$

2. The *mutual information* between p and q *given* s is

$$I(p : q \mid s) = K(q \mid s) - K(q \mid p, s).$$

The following properties of mutual information are well known [17].

Theorem 4.1. *Let $p \in \mathbb{Q}^m$ and $q \in \mathbb{Q}^n$.*

1. $I(p, K(p) : q) = K(p) + K(q) - K(p, q) + O(1)$.
2. $I(p, K(p) : q) = I(q, K(q) : p) + O(1)$.
3. $I(p : q) \leq \min\{K(p), K(q)\} + O(1)$.

(Each of the properties 1 and 2 above is sometimes called *symmetry of mutual information.*)

Mutual information between points in Euclidean space at a given precision is now defined as follows.

Definition. The *mutual information* of $x \in \mathbb{R}^n$ and $y \in \mathbb{R}^t$ at *precision* $r \in \mathbb{N}$ is

$$I_r(x : y) = \min\{I(q_x : q_y) \mid q_x \in B_{2^{-r}}(x) \cap \mathbb{Q}^n \text{ and } q_y \in B_{2^{-r}}(y) \cap \mathbb{Q}^t\}.$$

As noted in the introduction, the role of the minimum in the above definition is to eliminate "spurious" information that points $q_x \in B_{2^{-r}} \cap \mathbb{Q}^n$ and $q_y \in B_{2^{-r}}(y) \cap \mathbb{Q}^t$ might share for reasons not forced by their proximities to x and y, respectively.

Notation. We also use the quantity

$$J_r(x : y) = \min\{I(q_x : q_y) \mid p_x \text{ is a K-minimizer of } B_{2^{-r}}(x) \text{ and } p_y \text{ is a K–minimizer of } B_{2^{-r}}(y)\}.$$

Although $J_r(x : y)$, having two "layers of minimization", is somewhat more involved than $I_r(x : y)$, one can imagine using it as the definition of mutual information. In fact, for all $x, y \in \mathbb{R}$, $J_r(x : y)$ does not differ greatly from $I_r(x : y)$. We next develop machinery for proving this useful fact, which is Theorem 4.8 below.

Lemma 4.2. *There is a constant $c \in \mathbb{N}$ such that, for any $r \in \mathbb{N}$, open ball $B \subseteq \mathbb{R}^n$ of radius 2^{-r}, and $q \in B \cap \mathbb{Q}^n$,*

$$|\{p' \in B_{2^{1-r}}(q) \cap \mathbb{Q}^n \mid K(p') \leq K(B)\}| \leq 2^{K(r)+2K(r-1)+c}.$$

Proof. Let B be centered at $x \in \mathbb{R}^n$. If $p_q \in \mathbb{Q}^n$ is a K-minimizer of $B_{2^{1-r}}(q)$, then $p_q \in B_{2^{2-r}}(x)$. By Lemma 3.8,

$$\begin{aligned} K(B) &\leq K(p_q) + K(r) + c \\ &= K(B_{2^{1-r}}(q)) + K(r) + c, \end{aligned}$$

where $c = K(2) + K(n) + 2n(\lceil \frac{1}{2} \log n \rceil + 5) + 1 + c'$ for some constant c'. This inequality implies that any K-minimizer of B is also a $K(r)+c$-approximate K-minimizer of $B_{2^{1-r}}(q)$. Therefore, by Lemma 3.6,

$$\begin{aligned} |\{p' \in B_{2^{1-r}}(q) \cap \mathbb{Q}^n \mid K(p') \leq K(B)\}| &\leq |\{p' \in B_{2^{1-r}}(q) \cap \mathbb{Q}^n \mid K(p') \leq K(B_{2^{1-r}}(q)) + K(r) + c\}| \\ &\leq 2^{K(r)+2K(r-1)+c}. \end{aligned}$$

□

Lemma 4.3. *For all $x \in \mathbb{R}^n$, $q \in \mathbb{Q}^t$, and $q_x, p_x \in B_{2^{-r}}(x) \cap \mathbb{Q}^n$ where p_x is a K-minimizer of $B_{2^{-r}}(x)$,*

$$K(q \mid q_x) \leq K(q \mid p_x) + K(K(p_x)) + o(r).$$

Proof. Let M be a self-delimiting Turing machine that takes programs of the form $\pi = \langle \pi_1\pi_2\pi_3 0^{|s_i|}1s_i, q\rangle$, where $U(\pi_1, p) = q' \in \mathbb{Q}^t$, $U(\pi_2) = K(p)$, $U(\pi_3) = r \in \mathbb{N}$, and $i \in \mathbb{N}$. M runs π_2 and π_3 on U to obtain $K(p)$ and r, performs a systematic search for the i^{th} discovered element of $\{p' \in B_{2^{1-r}}(q) \cap \mathbb{Q}^n \mid K(p') \leq K(p)\}$, and outputs $U(\langle \pi_1, p_i\rangle)$. Therefore,

$$M(\pi) = U(\langle \pi_1, p_i\rangle). \tag{4.1}$$

Let c_M be an optimality constant for M.

Assume the hypothesis, and let $\pi = \langle \pi_1\pi_2\pi_3 0^{|s_i|}1s_i, q_x\rangle$, where π_1 is a minimum-length program for q when given p_x, π_2 is a minimum-length program for $K(p_x)$, π_3 is a minimum-length program for r, and i is an index for p_x in the set $\{p' \in B_{2^{1-r}}(q_x) \cap \mathbb{Q}^n \mid K(p') \leq K(p_x)\}$. By (4.1), we have $M(\pi) = U(\langle \pi_1, p_x\rangle) = q$. Therefore, by Lemma 4.2 and optimality,

$$\begin{aligned} K(q \mid q_x) &\leq K_M(q \mid q_x) + c_M \\ &\leq |\pi_1\pi_2\pi_3 0^{|s_i|}1s_i| + c_M \\ &= K(q \mid p_x) + K(K(p_x)) + K(r) + 2|s_i| + 1 + c_M \\ &\leq K(q \mid p_x) + K(K(p_x)) + K(r) + 2\log|\{p' \in B_{2^{1-r}}(q_x) \cap \mathbb{Q}^n \mid K(p') \leq K(p_x)\}| + 1 + c_M \\ &\leq K(q \mid p_x) + K(K(p_x)) + K(r) + 2(K(r) + 2K(r-1) + c) + 1 + c_M \\ &= K(q \mid p_x) + K(K(p_x)) + o(r). \end{aligned}$$

□

By Lemma 4.3 and Observation 3.7 we have the following.

Corollary 4.4. *Let $x \in \mathbb{R}^n$. If $q_x \in B_{2^{-r}}(x) \cap \mathbb{Q}^n$ and $p_x \in \mathbb{Q}^n$ is a K-minimizer of $B_{2^{-r}}(x)$, then $K(p_x \mid q_x) = o(r)$.*

Lemma 4.5. *Let $x \in \mathbb{R}^n$ and $y \in \mathbb{R}^t$. If $p_x \in B_{2^{-r}}(x)$ and $q_y, p_y \in B_{2^{-r}}(y)$ where p_x is a K-minimizer for $B_{2^{-r}}(x)$ and p_y is a K-minimizer for $B_{2^{-r}}(y)$, then*

$$K(p_x \mid q_y, K(q_y)) \leq K(p_x \mid p_y, K(p_y)) + o(r).$$

Proof. By the triangle inequality for strings and Corollary 4.4,

$$\begin{aligned} K(p_x \mid q_y, K(q_y)) &\leq K(p_x \mid p_y, K(p_y)) + K(p_y \mid q_y, K(q_y)) + O(1) \\ &\leq K(p_x \mid p_y, K(p_y)) + K(p_y \mid q_y) + O(1) \\ &= K(p_x \mid p_y, K(p_y)) + o(r). \end{aligned}$$

□

The following lemma was inspired by Hammer et al. [11].

Lemma 4.6. *For all* $x, y, z \in \{0,1\}^*$,

$$K(z) - K(K(z)) - K(K(x)) \leq I(x:y) + K(z \mid x, K(x)) + K(z \mid y, K(y)) - K(z \mid \langle x, y\rangle, K(\langle x, y\rangle)) - I(x:y|z) + O(1).$$

Proof. By the well-known identity (2.1), obvious inequalities, and basic definitions.

$$\begin{aligned}
&K(z) - K(K(z)) - K(K(x)) \\
&= K(x) - K(x,y) - K(K(x)) + K(x,z) - K(x) + K(y,z) - K(x,y,z) \\
&\quad + K(x,y) + K(z) - K(z,y) - K(K(z)) + K(x,z,y) - K(x,z) + O(1) \\
&= -K(y \mid x, K(x)) - K(K(x)) + K(x,z) - K(x) + K(y,z) - K(x,y,z) \\
&\quad + K(x,y) - K(y \mid z, K(z)) - K(K(z)) + K(y \mid x, z, K(x,z)) + O(1) \\
&\leq K(y) - K(y \mid x) + K(x,z) - K(x) + K(y,z) - K(y) - K(x,y,z) + K(x,y) \\
&\quad - K(y \mid z) + K(y \mid x, z) + O(1) \\
&= I(x:y) + K(z \mid x, K(x)) + K(z \mid y, K(y)) - K(z \mid x, y, K(x,y)) - I(x:y \mid z) + O(1).
\end{aligned}$$

□

Corollary 4.7. *For all* $x, y, z \in \{0,1\}^*$,

$$I(x:y) \geq K(z) - K(z \mid x, K(x)) - K(z \mid y, K(y)) - K(K(x)) - K(K(z)) + O(1).$$

Theorem 4.8. *For all* $x \in \mathbb{R}^n$ *and* $y \in \mathbb{R}^t$,

$$I_r(x:y) = J_r(x:y) + o(r).$$

Proof. Let $q_x, p_x \in \mathbb{Q}^n$ and $q_y, p_y \in \mathbb{Q}^t$ where p_x is a K-minimizer of $B_{2^{-r}}(x)$, p_y is a K-minimizer of $B_{2^{-r}}(y)$, and $I(q_x : q_y) = I_r(x:y)$. By Lemma 4.3,

$$K(q_y) - K(q_y \mid p_x) \leq K(q_y) - K(q_y \mid q_x) + K(K(p_x)) + o(r).$$

Applying the definition of mutual information for rationals, we have

$$I(p_x : q_y) \leq I(q_x : q_y) + K(K(p_x)) + o(r),$$

which, by Corollary 4.7 and Observation 3.7, implies that

$$\begin{aligned}
I(q_x : q_y) &\geq K(p_x) - K(p_x \mid p_x, K(p_x)) - K(p_x \mid q_y, K(q_y)) + o(r) \\
&= K(p_x) - K(p_x \mid q_y, K(q_y)) + o(r).
\end{aligned}$$

By applying Lemma 4.5 and the definition of mutual information for rationals to the above inequality, we obtain

$$\begin{aligned}
I(q_x : q_y) &\geq K(p_x) - K(p_x \mid p_y, K(p_y)) + o(r) \\
&= I(p_y, K(p_y) : p_x) + o(r).
\end{aligned}$$

Thus, by Theorem 4.1,

$$\begin{aligned} I(q_x : q_y) &\geq I(p_x, K(p_x) : p_y) + o(r) \\ &\geq I(p_x : p_y) + o(r). \end{aligned}$$

The above inequality tells us that $I_r(x : y) = I(q_x : q_y) \geq I(p_x : p_y) + o(r) = J_r(x : y) + o(r)$. Also, by definition, $I_r(x : y) \leq J_r(x : y)$. □

Before discussing the properties of $I_r(x : y)$, we need one more lemma.

Lemma 4.9. *Let $x \in \mathbb{R}^n$, $y \in \mathbb{R}^t$, and $r \in \mathbb{N}$. If $p_x \in \mathbb{Q}^n$ is a K-minimizer of $B_{2^{-r}}(x)$ and $p_y \in \mathbb{Q}^t$ is a K-minimizer of $B_{2^{-r}}(y)$, then*

$$K(p_x, p_y) = K_r(x, y) + o(r).$$

Proof. By Corollary 4.4,

$$\begin{aligned} K_r(x, y) \leq K(p_x, p_y) &\leq K(p_y) + K(p_x \,|\, p_y) \\ &= K_r(y) + K(p_x \,|\, p_y) \\ &\leq K_r(x, y) + K(p_x \,|\, p_y) + O(1) \\ &= K_r(x, y) + o(r). \end{aligned}$$ □

The following characterization of algorithmic (Martin-Löf) randomness is well known.

Definition. A point $x \in \mathbb{R}^n$ is *random* if there is a constant $d \in \mathbb{N}$ such that, for all $r \in \mathbb{N}$,

$$K_r(x) \geq nr - d.$$

Two points $x \in \mathbb{R}^n$ and $y \in \mathbb{R}^t$ are *independently random* if the point $(x, y) \in \mathbb{R}^{n+t}$ is random.

We now establish the following useful properties of $I_r(x : y)$.

Theorem 4.10. *For all $x \in \mathbb{R}^n$ and $y \in \mathbb{R}^t$,*

1. $I_r(x : y) = K_r(x) + K_r(y) - K_r(x, y) + o(r)$.
2. $I_r(x : y) \leq \min\{K_r(x), K_r(y)\} + o(r)$.
3. *If x and y are independently random, then $I_r(x : y) = o(r)$.*
4. $I_r(x : y) = I_r(y : x) + o(r)$.

Proof. To prove the first statement, let $p_x \in \mathbb{Q}^n$ be a K-minimizer of $B_{2^{-r}}(x)$ and $p_y \in \mathbb{Q}^t$ be a K-minimizer of $B_{2^{-r}}(y)$. First, by Theorem 4.8,

$$\begin{aligned} I_r(x : y) &= J_r(x : y) + o(r) \\ &= I(p_x : p_y) + o(r) \\ &= K(p_y) - K(p_y \,|\, p_x) + o(r) \\ &\leq K(p_y) - K(p_y \,|\, p_x, K(p_x)) + o(r). \end{aligned}$$

By (2.1) and Lemma 4.9, this implies that

$$\begin{aligned} I_r(x:y) &\leq K(p_y) + K(p_x) - K(p_x, p_y) + o(r) \\ &= K_r(y) + K_r(x) - K_r(x, y) + o(r). \end{aligned}$$

Next we show that $I_r(x:y) \geq K_r(x) + K_r(y) - K_r(x,y) + o(r)$. By the above inequality,

$$\begin{aligned} I_r(x:y) &= K(p_y) - K(p_y \mid p_x) + o(r) \\ &\geq K(p_y) - K(p_y \mid p_x, K(p_x)) - K(K(p_x)) + o(r). \end{aligned}$$

Finally, by (2.1), Observation 3.7, and Lemma 4.9,

$$\begin{aligned} I_r(x:y) &\geq K(p_y) + K(p_x) - K(p_x, p_y) + o(r) \\ &\geq K_r(y) + K_r(x) - K_r(x, y) + o(r). \end{aligned}$$

We continue to the second statement. By 1,

$$\begin{aligned} I_r(x:y) &= K_r(x) + K_r(y) - K_r(x, y) + o(r) \\ &\leq K_r(x) + K_r(y) - K_r(y) + o(r) \\ &= K_r(x) + o(r). \end{aligned}$$

Likewise, $I_r(x:y) \leq K_r(y) + o(r)$. Therefore, $I_r(x:y) \leq \min\{K_r(x), K_r(y)\} + o(r)$.

We now prove the third statement. By 1,

$$\begin{aligned} I_r(x:y) &= K_r(x) + K_r(y) - K_r(x, y) + o(r) \\ &\leq K_r(x) + K_r(y) + K(r) - K_r(r, x, y) + o(r) \\ &\leq nr + tr + K(r) - (n+t)r + o(r) \\ &= o(r), \end{aligned}$$

where the last inequality is due to the premise that x and y are independently random and Observation 3.7.

Lastly, we prove the fourth statement. By 1 and Lemma 4.9,

$$\begin{aligned} I_r(x:y) &= K_r(x) + K_r(y) - K_r(x, y) + o(r) \\ &= K_r(x) + K_r(y) - K(p_x, p_y) + o(r) \\ &= K_r(x) + K_r(y) - K(p_y, p_x) + o(r) \\ &= K_r(x) + K_r(y) - K_r(y, x) + o(r) \\ &= I_r(y:x) + o(r). \end{aligned}$$

□

5 Mutual Dimension in Euclidean Space

We now define mutual dimensions between points in Euclidean space(s).

Definition. The *lower* and *upper mutual dimensions* between $x \in \mathbb{R}^n$ and $y \in \mathbb{R}^t$ are

$$mdim(x:y) = \liminf_{r\to\infty} \frac{I_r(x:y)}{r}$$

and

$$Mdim(x:y) = \limsup_{r\to\infty} \frac{I_r(x:y)}{r},$$

respectively.

With the exception of the data processing inequality, which we prove in section 5, the following theorem says that the mutual dimensions $mdim$ and $Mdim$ have the basic properties that any mutual information measure should have. (See, for example, [2].)

Theorem 5.1. *For all $x \in \mathbb{R}^n$ and $y \in \mathbb{R}^t$, the following hold.*

1. $dim(x) + dim(y) - Dim(x,y) \leq mdim(x:y) \leq Dim(x) + Dim(y) - Dim(x,y)$.
2. $dim(x) + dim(y) - dim(x,y) \leq Mdim(x:y) \leq Dim(x) + Dim(y) - dim(x,y)$.
3. $mdim(x:y) \leq \min\{dim(x), dim(y)\}$, $Mdim(x:y) \leq \min\{Dim(x), Dim(y)\}$.
4. $0 \leq mdim(x:y) \leq Mdim(x:y) \leq \min\{n,t\}$.
5. *If x and y are independently random, then* $Mdim(x:y) = 0$.
6. $mdim(x:y) = mdim(y:x)$, $Mdim(x:y) = Mdim(y:x)$.

Proof. To prove the first statement, we use Theorem 4.10 and basic properties of lim sup and lim inf. First we show that $mdim(x:y) \geq dim(x) + dim(y) - Dim(x,y)$.

$$\begin{aligned}
mdim(x:y) &= \liminf_{r\to\infty} \frac{I_r(x:y)}{r} \\
&= \liminf_{r\to\infty} \frac{K_r(x) + K_r(y) - K_r(x,y) + o(r)}{r} \\
&\geq \liminf_{r\to\infty} \frac{K_r(x)}{r} + \liminf_{r\to\infty} \frac{K_r(y)}{r} + \liminf_{r\to\infty} \frac{-K_r(x,y)}{r} + \liminf_{r\to\infty} \frac{o(r)}{r} \\
&= dim(x) + dim(y) - \limsup_{r\to\infty} \frac{K_r(x,y)}{r} \\
&= dim(x) + dim(y) - Dim(x,y).
\end{aligned}$$

Next we show that $mdim(x:y) \leq Dim(x) + Dim(y) - Dim(x,y)$.

$$\begin{aligned}
mdim(x:y) &= Dim(x) + Dim(y) - Dim(x) - Dim(y) + mdim(x:y) \\
&= Dim(x) + Dim(y) - \Big(\limsup_{r\to\infty} \frac{K_r(x)}{r} + \limsup_{r\to\infty} \frac{K_r(y)}{r} + \limsup_{r\to\infty} \frac{-I_r(x:y)}{r} \Big) \\
&\leq Dim(x) + Dim(y) - \limsup_{r\to\infty} \frac{K_r(x) + K_r(y) - K_r(x) - K_r(y) + K_r(x,y) + o(r)}{r} \\
&= Dim(x) + Dim(y) - Dim(x,y).
\end{aligned}$$

The proof of the second statement is similar to the first. The third statement follows immediately from Theorem 4.10 and the fact that $\liminf_{r\to\infty} \min\{K_r(x), K_r(y)\} \leq \min\{\liminf_{r\to\infty} K_r(x), \liminf_{r\to\infty} K_r(y)\}$. The fourth statement follows from the third and the fact that, for all $x \in \mathbb{R}^n$, $Dim(x) \leq n$. Finally, both the fifth and sixth statements follow immediately from Theorem 4.10. □

6 Data Processing Inequalities

Our objectives in this section are to prove data processing inequalities for lower and upper mutual dimensions in Euclidean space.

The following result is the main theorem of this paper. The meaning and necessity of the Lipschitz hypothesis are explained in the introduction.

Theorem 6.1 (data processing inequality)**.** *If $f : \mathbb{R}^n \to \mathbb{R}^t$ is computable and Lipschitz, then, for all $x \in \mathbb{R}^n$ and $y \in \mathbb{R}^t$,*

$$mdim(f(x) : y) \leq mdim(x : y)$$

and

$$Mdim(f(x) : y) \leq Mdim(x : y).$$

We in fact prove a stronger result.

Definition. A *modulus (of uniform continuity)* for a function $f : \mathbb{R}^n \to \mathbb{R}^k$ is a nondecreasing function $m : \mathbb{N} \to \mathbb{N}$ such that, for all $x, y \in \mathbb{R}^n$ and $r \in \mathbb{N}$,

$$|x - y| \leq 2^{-m(r)} \Rightarrow |f(x) - f(y)| \leq 2^{-r}.$$

Note that it is well known that a function is uniformly continuous if and only if it has a modulus of uniform continuity.

Lemma 6.2. *For all strings $x, y, z \in \{0,1\}^*$ and all partial computable functions $f : \{0,1\}^* \times \{0,1\}^*$,*

$$K(y \mid x) \leq K(y \mid f(x, z)) + K(z) + O(1).$$

Proof. Let M be a self-delimiting Turing machine such that if $U(\pi_1, f(x, z)) = y$, $U(\pi_2) = z$, and π_3 is a program for f where $x, y, z \in \{0,1\}^*$ and $f : \{0,1\}^* \times \{0,1\}^*$ is a partial computable function, then

$$M(\pi_1\pi_2\pi_3, x) = y. \tag{6.1}$$

Assume the hypothesis, and let $\pi = \pi_1\pi_2\pi_3$ where π_1 is a minimum-length program for y given $f(x, z)$, π_2 is a minimum-length program for z, and π_3 is a minimum-length program for f. Therefore, by (6.1), we have $M(\pi, x) = y$. By optimality,

$$\begin{aligned} K(y \mid x) &\leq K_M(y \mid x) + c_M \\ &\leq |\pi| + c_M \\ &= K(y \mid f(x, z)) + K(z) + K(f) + c_M \\ &= K(y \mid f(x, z)) + K(z) + O(1). \end{aligned}$$

□

Lemma 6.3. *If $f : \mathbb{R}^n \to \mathbb{R}^k$ is computable and $m : \mathbb{N} \to \mathbb{N}$ is a computable modulus for f, then for every $x \in \mathbb{R}^n$, $y \in \mathbb{R}^t$,*

$$I_r(f(x) : y) \leq I_{m(r+1)}(x : y) + o(r).$$

Proof. Let $q_x \in \mathbb{Q}^n$ and $q_y \in \mathbb{Q}^t$ such that $I_{m(r+1)}(x : y) = I(q_x : q_y)$. Because $|x - q_x| \leq 2^{-m(r+1)}$, where m is a modulus for f, we know that $|f(x) - f(q_x)| \leq 2^{-(r+1)}$. Also, since f is computable, there exists an oracle Turing machine M that uses an oracle q_x such that $|M^{q_x}(r) - f(q_x)| \leq 2^{-r}$. Let $h : \mathbb{N} \times \mathbb{Q}^n \to \mathbb{Q}^k$ be a function such that $h(q_x, r) = M^{q_x}(r + 1)$. Observe that

$$\begin{aligned}
|h(q_x, r + 1) - f(x)| &\leq |f(x) - f(q_x)| + |f(q_x) - h(q_x, r)| \\
&\leq 2^{-(r+1)} + 2^{-(r+1)} \\
&= 2^{-r}.
\end{aligned}$$

From this and Lemma 6.2, it follows that

$$\begin{aligned}
I_r(f(x) : y) &\leq I(M^{q_x}(r + 1) : q_y) \\
&= I(h(q_x, r) : q_y) \\
&\leq I(q_x : q_y) + K(r) + O(1) \\
&= I_{m(r+1)}(x : y) + o(r).
\end{aligned}$$

□

Lemma 6.4 (modulus processing lemma)**.** *If $f : \mathbb{R}^n \to \mathbb{R}^k$ is computable and m is a computable modulus for f, then for all $x \in \mathbb{R}^n$ and $y \in \mathbb{R}^t$,*

$$mdim(f(x) : y) \leq mdim(x : y)\left(\limsup_{r\to\infty} \frac{m(r+1)}{r}\right)$$

and

$$Mdim(f(x) : y) \leq Mdim(x : y)\left(\limsup_{r\to\infty} \frac{m(r+1)}{r}\right).$$

Proof. By Lemma 6.3, we have

$$\begin{aligned}
mdim(f(x) : y) &\leq \liminf_{r\to\infty} \frac{I_{m(r+1)}(x : y)}{r} \\
&= \liminf_{r\to\infty} \left(\frac{I_{m(r+1)}(x : y)}{m(r+1)} \cdot \frac{m(r+1)}{r}\right) \\
&\leq mdim(x : y)\left(\limsup_{r\to\infty} \frac{m(r+1)}{r}\right).
\end{aligned}$$

A similar proof can be given for $Mdim$. □

Theorem 6.1 follows immediately from Lemma 6.4 and the following well-known observation.

Observation 6.5. *A function $f : \mathbb{R}^n \to \mathbb{R}^k$ is Lipschitz if and only if there exists $s \in \mathbb{N}$ such that $m(r) = r + s$ is a modulus for f.*

We can derive a similar observation for Hölder functions. (Recall the definition of Hölder functions given in the introduction.)

Observation 6.6. *If a function $f : \mathbb{R}^n \to \mathbb{R}^k$ is Hölder with exponent α, then there exists $s \in \mathbb{N}$ such that $m(r) = \lceil \frac{1}{\alpha}(r + s) \rceil$ is a modulus for f.*

We can derive the following fact from Observation 6.6 and the modulus processing lemma.

Corollary 6.7. *If $f : \mathbb{R}^n \to \mathbb{R}^k$ is computable and Hölder with exponent α, then, for all $x \in \mathbb{R}^n$ and $y \in \mathbb{R}^t$,*

$$mdim(f(x) : y) \leq \frac{1}{\alpha} mdim(x : y)$$

and

$$Mdim(f(x) : y) \leq \frac{1}{\alpha} Mdim(x : y).$$

7 Reverse Data Processing Inequalities

In this section we develop reverse versions of the data processing inequalities from section 6.

Notation. Let $n \in \mathbb{Z}^+$.

1. $[n] = \{1, \cdots, n\}$.
2. For $S \subseteq [n]$, $x \in \mathbb{R}^{|S|}$, $y \in \mathbb{R}^{n-|S|}$, the string

   $$x *_S y \in \mathbb{R}^n$$

   is obtained by placing the components of x into the positions in S (in order) and the components of y into the positions in $[n] \setminus S$ (in order).
3. For each $x = (x_1, x_2, \ldots, x_n) \in \mathbb{R}^n$, let $x_{(i,j)} = (x_i, x_{i+1}, \ldots, x_j)$ for every $i, j \in \mathbb{N}$ such that $i \leq j \leq n$.

Definition. Let $f : \mathbb{R}^n \to \mathbb{R}^k$.

1. f is *co-Lipschitz* if there is a real number $c > 0$ such that for all $x, y \in \mathbb{R}^n$,

   $$|f(x) - f(y)| \geq c|x - y|.$$

2. f is *bi-Lipschitz* if f is both Lipschitz and co-Lipschitz.
3. For $S \subseteq [n]$, f is *S-co-Lipschitz* if there is a real number $c > 0$ such that, for all $u, v \in \mathbb{R}^{|S|}$ and $y \in \mathbb{R}^{n-|S|}$,

   $$|f(u *_S y) - f(v *_S y)| \geq c|u - v|.$$

4. For $i \in [n]$, f is *co-Lipschitz in its i^{th} argument* if f is $\{i\}$-co-Lipschitz.

Note that f is $[n]$-co-Lipschitz if and only if f is co-Lipschitz.

Example. The function $f : \mathbb{R}^n \to \mathbb{R}$ defined by

$$f(x_1, \cdots, x_n) = x_1 + \cdots + x_n$$

is S-co-Lipschitz if and only if $|S| \leq 1$. In particular, if $n \geq 2$, then f is co-Lipschitz in every argument, but f is not co-Lipschitz.

We next relate co-Lipschitz conditions to moduli.

Definition. Let $f : \mathbb{R}^n \to \mathbb{R}^k$.

1. An *inverse modulus* for f is a nondecreasing function $m' : \mathbb{N} \to \mathbb{N}$ such that, for all $x, y \in \mathbb{R}^n$ and $r \in \mathbb{N}$,
$$|f(x) - f(y)| \leq 2^{-m'(r)} \Rightarrow |x - y| \leq 2^{-r}.$$

2. Let $S \subseteq [n]$. An S-*inverse modulus* for f is a nondecreasing function $m' : \mathbb{N} \to \mathbb{N}$ such that, for all $u, v \in \mathbb{R}^{|S|}$, all $y \in \mathbb{R}^{n-|S|}$, and all $r \in \mathbb{N}$,
$$|f(u *_S y) - f(v *_S y)| \leq 2^{-m'(r)} \Rightarrow |u - v| \leq 2^{-r}.$$

3. Let $i \in [n]$. An *inverse modulus* for f *in its* i^{th} *argument* is an $\{i\}$-inverse modulus for f.

Observation 7.1. *Let* $f : \mathbb{R}^n \to \mathbb{R}^k$ *and* $S \subseteq [n]$.

1. f *is* S*-co-Lipschitz if and only if there is a positive constant* $t \in \mathbb{N}$ *such that* $m'(r) = r + t$ *is an* S*-inverse modulus of* f.

2. f *is co-Lipschitz if and only if there is a positive constant* $t \in \mathbb{N}$ *such that* $m'(r) = r + t$ *is an inverse modulus of* f.

Definition. Let $f : \mathbb{R}^n \to \mathbb{R}^t$ and $S \subseteq [n]$. We say that f is S-*injective* if, for all $x, y \in \mathbb{R}^n$ and $z \in \mathbb{R}^{n-|S|}$,
$$f(x *_S z) = f(y *_S z) \Rightarrow x = y.$$

Note f is injective if and only if f is $[n]$-injective.

Definition. Let $f : \mathbb{R}^n \to \mathbb{R}^t$ be a function and $S \subseteq [n]$ such that $n \in \mathbb{N}$. An S-*left inverse* of f is a partial function $g : \mathbb{R}^t \times \mathbb{R}^{n-|S|} \to \mathbb{R}^{|S|}$ such that, for all $x \in \mathbb{R}^{|S|}$ and $y \in \mathbb{R}^t \times \mathbb{R}^{n-|S|}$,
$$g(f(x *_S y), y) = x.$$

It is easy to prove that f has an S-left inverse if and only if f is S-injective.

Lemma 7.2. *If* $f : \mathbb{R}^n \to \mathbb{R}^t$ *has an* S*-inverse modulus* m'*, then* f *is* S*-injective and* m' *is a modulus for any* S*-left inverse of* f.

Proof. Let $m' : \mathbb{N} \to \mathbb{N}$ be an S-inverse modulus for f, $x, y \in \mathbb{R}^{|S|}$ and $z \in \mathbb{R}^{n-|S|}$, then, if $f(x *_S z) = f(y *_S z)$,
$$|f(x *_S z) - f(y *_S z)| \leq 2^{-m'(r)},$$
for all $r \in \mathbb{N}$, which implies that
$$|x - y| \leq 2^{-r}.$$

Therefore, $x = y$ and f is S-injective.

Let $g : \mathbb{R}^t \times \mathbb{R}^{n-|S|} \to \mathbb{R}^{|S|}$ be an S-left inverse of f. Let $x, y \in dom\, g$ and $r \in \mathbb{N}$ such that $x = (f(u *_S w), w)$ and $y = (f(v *_S z), z)$, where $u, v \in \mathbb{R}^{|S|}$ and $w, z \in \mathbb{R}^{n-|S|}$. Assume that $|x - y| \leq 2^{-m'(r)}$, then

$$\begin{aligned} & |f(g(f(u *_S w), w) *_S w) - f(g(f(v *_S z), z) *_S z)| \\ = & |f(u *_S w) - f(v *_S z)| \\ \leq & |(f(u *_S w), w) - (f(v *_S z), z)| \\ = & |x - y| \\ \leq & 2^{-m'(r)}. \end{aligned}$$

So, $|g(f(u *_S w), w) - g(f(v *_S z), z)| \leq 2^{-r}$, and

$$\begin{aligned} |g(x) - g(y)| &= |g(f(u *_S w), w) - g(f(v *_S z), z)| \\ &\leq 2^{-r}. \end{aligned}$$

Therefore, m' is a modulus for g. □

Lemma 7.3. *If $f : \mathbb{R}^n \to \mathbb{R}^t$ is a computable and uniformly continuous function that has a computable S-inverse modulus m', then f has a computable S-left inverse.*

Proof. Assume the hypothesis. Since f is computable and uniformly continuous, there exist a modulus m for f and an oracle Turing machine M_f such that, for every $x \in \mathbb{R}^n$, $r \in \mathbb{N}$, and every oracle h_x for x,

$$|M_f^{h_x}(r) - f(x)| \leq 2^{-r}. \tag{7.1}$$

Define $g : \mathbb{R}^t \times \mathbb{R}^{n-|S|} \to \mathbb{R}^{|S|}$ by

$$g(z) = \begin{cases} x & \text{if } z = (f(x *_S y), y), \\ \text{undefined} & \text{otherwise} \end{cases},$$

where $x \in \mathbb{R}^{|S|}$, $y \in \mathbb{R}^{n-|S|}$, and $z \in \mathbb{R}^t \times \mathbb{R}^{n-|S|}$.

We now show that g is computable. Let $z = (f(x *_S y), y) \in dom\, g$ and h_z be an oracle for z such that, for all $r \in \mathbb{N}$,

$$|h_z(r) - z| \leq 2^{-r}. \tag{7.2}$$

First we show that, for any $r \in \mathbb{N}$, there exist a rational $q \in \mathbb{Q}^{|S|}$ and an oracle h_{qy} for $q *_S y$ such that

$$|M_f^{h_{qy}}(m'(r) + 3) - h_z(m'(r) + 3)| \leq 2^{-(m'(r)+1)}.$$

Let $q \in \mathbb{Q}^{|S|}$ such that $|q *_S y - x *_S y| \leq 2^{-(m(m'(r)+2))}$, and let $h_{qy}(r) = q *_S h_z(r)_{(t+1, t+n-|S|)}$ be an oracle for $q *_S y$. Therefore,

$$|f(q *_S y) - f(x *_S y)| \leq 2^{-(m'(r)+2)}. \tag{7.3}$$

By (7.1), (7.2), (7.3),

$$
\begin{aligned}
&|M_f^{h_{qy}}(m'(r)+3) - h_z(m'(r)+3)_{(1,t)}| \\
&= |M_f^{h_{qy}}(m'(r)+3) - f(q *_S y) + f(q *_S y) - f(x *_S y) + f(x *_S y) - h_z(m'(r)+3)_{(1,t)}| \\
&\le |M_f^{h_{qy}}(m'(r)+3) - f(q *_S y)| + |f(q *_S y) - f(x *_S y)| + |h_z(m'(r)+3)_{(1,t)} - f(x *_S y)| \\
&\le 2^{-(m'(r)+3)} + 2^{-(m'(r)+2)} + 2^{-(m'(r)+3)} \\
&= 2^{-(m'(r)+1)}.
\end{aligned}
$$

Let M_g be a Turing machine equipped with oracle h_z. Given an input $r \in \mathbb{N}$, M_g searches for and outputs a rational $q_x \in \mathbb{Q}^{|S|}$ such that

$$
|M_f^{h_{q_x y}}(m'(r)+3) - h_z(m'(r)+3)_{(1,t)}| \le 2^{-(m'(r)+1)}, \tag{7.4}
$$

where $h_{q_x y} = q_x *_S h_z(r)_{(t+1,t+n-|S|)}$ is an oracle for $q_x *_S y$. We now show that $|M_g^{h_z}(r) - g(z)| \le 2^{-r}$. By (7.1), (7.2), (7.4),

$$
\begin{aligned}
&|f(M_g^{h_z}(r) *_S y) - f(x *_S y)| \\
&= |f(q_x *_S y) - f(x *_S y)| \\
&= |f(q_x *_S y) - M_f^{h_{q_x y}}(m'(r)+3) + M_f^{h_{q_x y}}(m'(r)+3) - h_z(m'(r)+3)_{(1,t)} + h_z(m'(r)+3)_{(1,t)} - f(x *_S y)| \\
&\le |f(q_x *_S y) - M_f^{h_{q_x y}}(m'(r)+3)| + |M_f^{h_{q_x y}}(m'(r)+3) - h_z(m'(r)+3)_{(1,t)}| + |h_z(m'(r)+3)_{(1,t)} - f(x *_S y)| \\
&\le 2^{-(m'(r)+3)} + 2^{-(m'(r)+1)} + 2^{-(m'(r)+3)} \\
&= 2^{-(m'(r)+2)} + 2^{(m'(r)+1)} \\
&< 2^{-m'(r)}.
\end{aligned}
$$

Since m' is an S-inverse modulus for f, we have

$$
\begin{aligned}
|M_g^{h_z}(r) - g(z)| &= |M_g^{h_z}(r) - x| \\
&\le 2^{-r}.
\end{aligned}
$$

Therefore, g is a computable S-left inverse of f. □

Lemma 7.4 (reverse modulus processing lemma)**.** *If $f : \mathbb{R}^n \to \mathbb{R}^k$ is a computable and uniformly continuous function, and m' is a computable S-inverse modulus for f, then, for all $S \subseteq [n]$, $x \in \mathbb{R}^{|S|}$, $y \in \mathbb{R}^t$, and $z \in \mathbb{R}^{n-|S|}$,*

$$
mdim(x : y) \le mdim((f(x *_S z), z) : y)\left(\limsup_{r\to\infty} \frac{m'(r+1)}{r}\right)
$$

and

$$
Mdim(x : y) \le Mdim((f(x *_S z), z) : y)\left(\limsup_{r\to\infty} \frac{m'(r+1)}{r}\right).
$$

Proof. Assume the hypothesis. By Lemmas 7.2 and 7.3, there exists a computable and uniformly continuous function g that is an S-left inverse of f and m' is a modulus for g. Then, for all $S \subseteq [n]$,

$x \in \mathbb{R}^{|S|}$, $y \in \mathbb{R}^t$, and $z \in \mathbb{R}^{n-|S|}$,

$$mdim(x : y) = mdim(g(f(x *_S z), z) : y).$$

Therefore, by Lemma 6.4, we have

$$mdim(x : y) \leq mdim(f(x *_S z), z : y)\left(\limsup_{r \to \infty} \frac{m'(r+1)}{r} \right).$$

A similar proof can be given for $Mdim$. □

By Observation 7.1 and Lemma 7.4, we have the following.

Theorem 7.5 (reverse data processing inequality)**.** *If $S \subseteq [n]$ and $f : \mathbb{R}^n \to \mathbb{R}^k$ is computable and S-co-Lipschitz, then, for all $x \in \mathbb{R}^{|S|}$, $y \in \mathbb{R}^t$, and $z \in \mathbb{R}^{n-|S|}$,*

$$mdim(x : y) \leq mdim((f(x *_S z), z) : y)$$

and

$$Mdim(x : y) \leq Mdim((f(x *_S z), z) : y).$$

Definition. Let $f : \mathbb{R}^n \to \mathbb{R}^k$ and $0 < \alpha \leq 1$.

1. f is *co-Hölder with exponent* α if there is a real number $c > 0$ such that, for all $x, y \in \mathbb{R}^n$,

$$|x - y| \leq c|f(x) - f(y)|^\alpha.$$

2. For $S \subseteq [n]$, f is *S-co-Hölder with exponent* α if there is a real number $c > 0$ such that, for all $u, v \in \mathbb{R}^{|S|}$ and $y \in \mathbb{R}^{n-|S|}$,

$$|u - v| \leq c|f(u *_S y) - f(v *_S y)|^\alpha.$$

Observation 7.6. *Let $f : \mathbb{R}^n \to \mathbb{R}^k$ and $S \subseteq [n]$.*

1. *If f is S-co-Hölder with exponent α, then there exists $t \in \mathbb{N}$ such that $m'(r) = \lceil \frac{1}{\alpha}(r + t) \rceil$ is an S-inverse modulus of f.*
2. *If f is co-Hölder with exponent α, then there exists $t \in \mathbb{N}$ such that $m'(r) = \lceil \frac{1}{\alpha}(r + t) \rceil$ is an inverse modulus of f.*

The next corollary follows from the reverse modulus processing lemma and Observation 7.6.

Corollary 7.7. *If $S \subseteq [n]$ and $f : \mathbb{R}^n \to \mathbb{R}^k$ is computable and S-co-Hölder with exponent α, then, for all $x \in \mathbb{R}^{|S|}$, $y \in \mathbb{R}^t$, and $z \in \mathbb{R}^{n-|S|}$,*

$$mdim(x : y) \leq \frac{1}{\alpha} mdim((f(x *_S z), z) : y)$$

and

$$Mdim(x : y) \leq \frac{1}{\alpha} Mdim((f(x *_S z), z) : y).$$

8 Data Processing Applications

In this section we use the data processing inequalities and their reverses to investigate how certain functions on Euclidean space preserve or predictably transform mutual dimensions.

Theorem 8.1 (mutual dimension conservation inequality)**.** *If $f : \mathbb{R}^n \to \mathbb{R}^k$ and $g : \mathbb{R}^t \to \mathbb{R}^l$ are computable and Lipschitz, then, for all $x \in \mathbb{R}^n$ and $y \in \mathbb{R}^t$,*

$$mdim(f(x) : g(y)) \leq mdim(x : y)$$

and

$$Mdim(f(x) : g(y)) \leq Mdim(x : y).$$

Proof. The conclusion follows from Theorem 5.1 and the data processing inequality.

$$\begin{aligned} mdim(f(x) : g(y)) &\leq mdim(x : g(y)) \\ &= mdim(g(y) : x) \\ &\leq mdim(y : x) \\ &= mdim(x : y). \end{aligned}$$

A similar argument can be given for $Mdim(f(x) : g(y)) \leq Mdim(x : y)$. □

Theorem 8.2 (reverse mutual dimension conservation inequality)**.** *Let $S_1 \subseteq [n]$ and $S_2 \subseteq [t]$. If $f : \mathbb{R}^n \to \mathbb{R}^k$ is computable and S_1-co-Lipschitz, and $g : \mathbb{R}^t \to \mathbb{R}^l$ is computable and S_2-co-Lipschitz, then, for all $x \in \mathbb{R}^{|S_1|}$, $y \in \mathbb{R}^{|S_2|}$, $w \in \mathbb{R}^{n-|S_1|}$, and $z \in \mathbb{R}^{t-|S_2|}$,*

$$mdim(x : y) \leq mdim((f(x *_S w), w) : (g(y *_S z), z))$$

and

$$Mdim(x : y) \leq Mdim((f(x *_S w), w) : (g(y *_S z), z)).$$

Proof. The conclusion follows from Theorem 5.1 and the reverse data processing inequality.

$$\begin{aligned} mdim(x : y) &\leq mdim((f(x *_S w), w) : y) \\ &= mdim(y : (f(x *_S w), w)) \\ &\leq mdim((g(y *_S z), z) : (f(x *_S w), w)) \\ &= mdim((f(x *_S w), w) : (g(y *_S z), z)). \end{aligned}$$

A similar argument can be given for $Mdim(x : y) \leq Mdim((f(x *_S w), w) : (g(y *_S z), z))$. □

Corollary 8.3 (preservation of mutual dimension)**.** *If $f : \mathbb{R}^n \to \mathbb{R}^k$ and $g : \mathbb{R}^t \to \mathbb{R}^l$ are computable and bi-Lipschitz, then, for all $x \in \mathbb{R}^n$ and $y \in \mathbb{R}^t$,*

$$mdim(f(x) : g(y)) = mdim(x : y)$$

and

$$Mdim(f(x) : g(y)) = Mdim(x : y).$$

Corollary 8.4. *If $f : \mathbb{R}^n \to \mathbb{R}^k$ and $g : \mathbb{R}^t \to \mathbb{R}^l$ are computable and Hölder with exponents α and β, respectively, then, for all $x \in \mathbb{R}^n$ and $y \in \mathbb{R}^t$,*

$$mdim(f(x) : g(y)) \leq \frac{1}{\alpha\beta} mdim(x : y)$$

and

$$Mdim(f(x) : g(y)) \leq \frac{1}{\alpha\beta} Mdim(x : y).$$

Corollary 8.5. *Let $S_1 \subseteq [n]$ and $S_2 \subseteq [t]$. If $f : \mathbb{R}^n \to \mathbb{R}^k$ is computable and S_1-co-Hölder with exponent α, and $g : \mathbb{R}^t \to \mathbb{R}^l$ is computable and S_2-co-Hölder with exponent β, then, for all $x \in \mathbb{R}^{|S_1|}$, $y \in \mathbb{R}^{|S_2|}$, $w \in \mathbb{R}^{n-|S_1|}$, and $z \in \mathbb{R}^{t-|S_2|}$,*

$$mdim(x : y) \leq \frac{1}{\alpha\beta} mdim((f(x *_S w), w) : (g(y *_S z), z))$$

and

$$Mdim(x : y) \leq \frac{1}{\alpha\beta} Mdim((f(x *_S w), w) : (g(y *_S z), z)).$$

9 Conclusion

We expect mutual dimensions and the data processing inequalities to be useful for future research in computable analysis. We also expect the development of mutual dimensions in Euclidean spaces — highly structured spaces in which it is clear that $mdim$ and $Mdim$ are the right notions — to guide future explorations of mutual information in more challenging contexts.

Acknowledgments. We thank Elvira Mayordomo, A. Pavan, Giora Slutzki, and Jim Lathrop for useful discussions. We also thank three anonymous reviewers for useful suggestions.

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 of Computing*, 37(3):671–705, 2007.

[2] C.B. Bell. Mutual information and maximal correlation as measures of dependence. *Annals of Mathematical Statistics*, 33(2):587–595, 1962.

[3] Mark Braverman and Stephen Cook. Computing over the reals: foundations for scientific computing. *Notices of the American Mathematical Society*, 53(3):318–329, 2006.

[4] P. J. Couch, B. D. Daniel, and Timothy H. McNicholl. Computing space-filling curves. *Theory of Computing Systems*, 50(2):370–386, 2012.

[5] Thomas R. Cover and Joy A. Thomas. *Elements of Information Theory.* John Wiley & Sons, Inc., second edition, 2006.

[6] Randall Dougherty, Jack H. Lutz, Daniel R. Mauldin, and Jason Teutsch. Translating the Cantor set by a random real. *Transactions of the American Mathematical Society*, 366(6):3027–3041, 2014.

[7] Kenneth Falconer. *Fractal Geometry: Mathematical Foundations and Applications*. Wiley, second edition, 2003.

[8] Peter Gács. On the symmetry of algorithmic information. *Soviet Mathematics Doklady*, 15(5):1477–1480, 1974. Correction, Ibid., 15:1480, 1974.

[9] Xiaoyang Gu, Jack Lutz, and Elvira Mayordomo. Curves that must be retraced. *Information and Computation*, 209(6):992–1006, 2011.

[10] Xiaoyang Gu, Jack H. Lutz, and Elvira Mayordomo. Points on computable curves. In *Foundations of Computer Science*, pages 469–474. IEEE Computer Society, 2006.

[11] Daniel Hammer, Andrei E. Romashchenko, Alexander Shen, and Nikolai K. Vereshchagin. Inequalities for Shannon entropy and Kolmogorov complexity. *Journal of Computer System Sciences*, 60(2):442–464, 2000.

[12] John M. Hitchcock. Correspondence principles for effective dimensions. *Theory of Computing Systems*, 38(5):559–571, 2005.

[13] Bjørn Kjos-Hanssen and Anil Nerode. Effective dimension of points visited by Brownian motion. *Theoretical Computer Science*, 410(4-5):347–354, 2009.

[14] Ker-I Ko. *Complexity Theory of Real Functions*. Birkhäuser, first edition, 1991.

[15] Leonid A. Levin. On the notion of a random sequence. *Soviet Mathematics Doklady*, 14(5):1413–1416, 1973.

[16] Leonid A. Levin. Laws of information conservation (nongrowth) and aspects of the foundation of probability theory. *Problemy Peredachi Informatsii*, 10(3):30–35, 1974.

[17] Ming Li and Paul Vitányi. *An Introduction to Kolmogorov Complexity and Its Applications*. Springer, third edition, 2008.

[18] Jack H. Lutz. The dimension of individual strings and sequences. *Information and Computation*, 187(1):49–79, 2003.

[19] Jack H. Lutz and Elvira Mayordomo. Dimensions of points in self-similar fractals. *SIAM Journal on Computing*, 38(3):1080–1112, 2008.

[20] Jack H. Lutz and Klaus Weihrauch. Connectivity properties of dimension level sets. *Mathematical Logic Quarterly*, 54(5):483–491, 2008.

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

[22] Robert Rettinger and Xizhong Zheng. Points on computable curves of computable lengths. In *MFCS*, pages 736–743. Springer, 2009.

[23] Alexander Shen and Nikolai K. Vereshchagin. Logical operations and Kolmogorov complexity. *Theoretical Computer Science*, 271(1-2):125–129, 2002.

[24] Daniel Turetsky. Connectedness properties of dimension level sets. *Theoretical Computer Science*, 412(29):3598–3603, 2011.

[25] Alan M. Turing. On computable numbers, with an application to the Entscheidungsproblem. A correction. *Proceedings of the London Mathematical Society*, 43(2):544–546, 1937.

[26] Klaus Weihrauch. *Computable Analysis: An Introduction.* Springer, first edition, 2000.

ECCC ISSN 1433-8092

http://eccc.hpi-web.de