NSF-CBMS Research Conference
Algorithmic Fractal Dimensions

Lecture 3
Three great theories of information

Jack Lutz
Iowa State University

May 21, 2024
Drake University

Shannon: Information Theory

Shannon's 1948 *The Mathematical Theory of Communication* was a world-changing publication.

Shannon: Information Theory

Shannon's 1948 *The Mathematical Theory of Communication* was a world-changing publication.

Definition

- A subprobability measure on a nonempty, countable set X is a function $p : X \to [0, 1]$ such that

$$\sum_{x \in X} p(x) \leq 1. \qquad (*)$$

Shannon: Information Theory

Shannon's 1948 *The Mathematical Theory of Communication* was a world-changing publication.

Definition

- A subprobability measure on a nonempty, countable set X is a function $p : X \to [0, 1]$ such that

$$\sum_{x \in X} p(x) \leq 1. \qquad (*)$$

- A probability measure on X is a subprobability measure p that satisfies $(*)$ with equality.

Shannon: Information Theory

Shannon's 1948 *The Mathematical Theory of Communication* was a world-changing publication.

Definition

- A subprobability measure on a nonempty, countable set X is a function $p : X \to [0, 1]$ such that

$$\sum_{x \in X} p(x) \leq 1. \qquad (*)$$

- A probability measure on X is a subprobability measure p that satisfies $(*)$ with equality.

A discrete (sub)probability space is an ordered pair (X, p) as above.

Shannon: Information Theory

Shannon's 1948 *The Mathematical Theory of Communication* was a world-changing publication.

Definition

- A subprobability measure on a nonempty, countable set X is a function $p : X \to [0, 1]$ such that

$$\sum_{x \in X} p(x) \leq 1. \qquad (*)$$

- A probability measure on X is a subprobability measure p that satisfies $(*)$ with equality.

A discrete (sub)probability space is an ordered pair (X, p) as above.

If X is finite, then a discrete probability space (X, p) is also called an ensemble.

Shannon: Information Theory

Let (X, p) be a discrete subprobability space.

Shannon: Information Theory

Let (X, p) be a discrete subprobability space.

Definition

The Shannon self-information of a point $x \in X$ is

$$\mathcal{I}_p(x) = \log \frac{1}{p(x)}.$$

(All logarithms here are base-2.)

Shannon: Information Theory

Let (X, p) be a discrete subprobability space.

Definition

The Shannon self-information of a point $x \in X$ is

$$\mathcal{I}_p(x) = \log \frac{1}{p(x)}.$$

(All logarithms here are base-2.)

Intuition: The probability $p(x)$ suggests that x "acts like" one of $\frac{1}{p(x)}$ equiprobable points. $\mathcal{I}_p(x)$ is the number of bits needed to specify each of these $\frac{1}{p(x)}$ points.

Shannon: Information Theory

Definition

The Shannon entropy of (X, p) is

$$H(X) = E\mathcal{I}_p(x) = E \log \frac{1}{p(x)} = \sum_{x \in X} p(x) \log \frac{1}{p(x)}.$$

Shannon: Information Theory

Uniqueness of Entropy (Khinchin 1953)

Let $\mathcal{E}$ be the set of all ensembles (X, p). The Shannon entropy is, up to the base of the logarithm, the unique function $H : \mathcal{E} \to [0, \infty)$ with the following properties.

Shannon: Information Theory

Uniqueness of Entropy (Khinchin 1953)

Let $\mathcal{E}$ be the set of all ensembles (X, p). The Shannon entropy is, up to the base of the logarithm, the unique function $H : \mathcal{E} \to [0, \infty)$ with the following properties.

1. H is continuous

Shannon: Information Theory

Uniqueness of Entropy (Khinchin 1953)

Let $\mathcal{E}$ be the set of all ensembles (X, p). The Shannon entropy is, up to the base of the logarithm, the unique function $H : \mathcal{E} \to [0, \infty)$ with the following properties.

1. H is continuous
2. H is invariant under permutations

Shannon: Information Theory

Uniqueness of Entropy (Khinchin 1953)

Let $\mathcal{E}$ be the set of all ensembles (X, p). The Shannon entropy is, up to the base of the logarithm, the unique function $H : \mathcal{E} \to [0, \infty)$ with the following properties.

1. H is continuous
2. H is invariant under permutations
3. H is maximized by uniform probabilities

Shannon: Information Theory

Uniqueness of Entropy (Khinchin 1953)

Let $\mathcal{E}$ be the set of all ensembles (X, p). The Shannon entropy is, up to the base of the logarithm, the unique function $H : \mathcal{E} \to [0, \infty)$ with the following properties.

1. H is continuous
2. H is invariant under permutations
3. H is maximized by uniform probabilities
4. If (X, p) and (Y, q) are ensembles with $X \subseteq Y$ and $q \equiv p$ on X, then $H(Y, q) = H(X, p)$.

Shannon: Information Theory

Uniqueness of Entropy (Khinchin 1953)

Let $\mathcal{E}$ be the set of all ensembles (X, p). The Shannon entropy is, up to the base of the logarithm, the unique function $H : \mathcal{E} \to [0, \infty)$ with the following properties.

1. H is continuous
2. H is invariant under permutations
3. H is maximized by uniform probabilities
4. If (X, p) and (Y, q) are ensembles with $X \subseteq Y$ and $q \equiv p$ on X, then $H(Y, q) = H(X, p)$.
5. If (X, p) and (Y, q) are ensembles, then

$$H(X \times Y, p \times q) = H(X, p) + H(Y, q).$$

Definition

A set $A \subseteq \{0,1\}^*$ is prefix-free (or is a prefix set) if no element of A is a prefix of any other element of A.

Kolmogorov Complexity

Definition

A set $A \subseteq \{0,1\}^*$ is prefix-free (or is a prefix set) if no element of A is a prefix of any other element of A.

Definition

A prefix Turing machine is a Turing machine M whose domain (set of inputs π such that $M(\pi)$ halts) is a prefix set.

Kolmogorov Complexity

Definition

A set $A \subseteq \{0,1\}^*$ is prefix-free (or is a prefix set) if no element of A is a prefix of any other element of A.

Definition

A prefix Turing machine is a Turing machine M whose domain (set of inputs π such that $M(\pi)$ halts) is a prefix set.

The theory of prefix Turing machines is much like the theory of Turing machines: There is a standard enumeration of $M_0, M_1, M_2, \ldots,$ of prefix TMs, there is a universal prefix TM U such that each $U(0^n 1x)$ simulates $M_n(x)$, etc.

Kolmogorov Complexity

Definition

A set $A \subseteq \{0,1\}^*$ is prefix-free (or is a prefix set) if no element of A is a prefix of any other element of A.

Definition

A prefix Turing machine is a Turing machine M whose domain (set of inputs π such that $M(\pi)$ halts) is a prefix set.

The theory of prefix Turing machines is much like the theory of Turing machines: There is a standard enumeration of $M_0, M_1, M_2, \ldots,$ of prefix TMs, there is a universal prefix TM U such that each $U(0^n 1x)$ simulates $M_n(x)$, etc.

We fix a universal prefix TM U.

Kolmogorov Complexity

The following is Levin's 1973 refinement of the notion of Kolmogorov complexity introduced by Solomonoff (1964), Kolmogorov (1965), and Chaitin (1966, 1969).

Recall that U is a fixed universal prefix Turing machine.

Kolmogorov Complexity

The following is Levin's 1973 refinement of the notion of Kolmogorov complexity introduced by Solomonoff (1964), Kolmogorov (1965), and Chaitin (1966, 1969).

Recall that U is a fixed universal prefix Turing machine.

Definition

The Kolmogorov complexity of a string $x \in \{0,1\}^*$ is

$$K(x) = \min\{|\pi| \mid \pi \in \{0,1\}^* \text{ and } U(\pi) = x\},$$

i.e., the minimum number of bits required to cause U to print x.

Kolmogorov Complexity

The following is Levin's 1973 refinement of the notion of Kolmogorov complexity introduced by Solomonoff (1964), Kolmogorov (1965), and Chaitin (1966, 1969).

Recall that U is a fixed universal prefix Turing machine.

Definition

The Kolmogorov complexity of a string $x \in \{0,1\}^*$ is

$$K(x) = \min\{|\pi| \mid \pi \in \{0,1\}^* \text{ and } U(\pi) = x\},$$

i.e., the minimum number of bits required to cause U to print x.

If $U(\pi) = x$, we call π a program for x.

Kolmogorov Complexity

Facts

Kolmogorov Complexity

Facts

- $K(x) \leq |x| + o(|x|)$

Kolmogorov Complexity

Facts

- $K(x) \leq |x| + o(|x|)$
- $K(x)$ is seldom much smaller than $|x|$.

Kolmogorov Complexity

Facts

- $K(x) \leq |x| + o(|x|)$
- $K(x)$ is seldom much smaller than $|x|$.
- Different choices of U agree on $K(x)$ to within small additive constants.

Kolmogorov Complexity

Facts

- $K(x) \leq |x| + o(|x|)$
- $K(x)$ is seldom much smaller than $|x|$.
- Different choices of U agree on $K(x)$ to within small additive constants.
- $K(x)$ is also called the algorithmic information content of x.

Kolmogorov Complexity

Facts

- $K(x) \leq |x| + o(|x|)$
- $K(x)$ is seldom much smaller than $|x|$.
- Different choices of U agree on $K(x)$ to within small additive constants.
- $K(x)$ is also called the algorithmic information content of x.

Theorem (Levin 1973)

A sequence $R \in \{0,1\}^\omega$ is random if and only if there is a constant $c \in \mathbb{N}$ such that, for all $x \sqsubseteq R$, $K(x) \geq |x| - c$.

Kraft's Inequality

If $A \subseteq \{0,1\}^*$ is prefix-free, then

$$\sum_{x \in A} 2^{-|x|} \leq 1. \qquad (*)$$

Kraft's Inequality

If $A \subseteq \{0,1\}^*$ is prefix-free, then

$$\sum_{x \in A} 2^{-|x|} \leq 1. \qquad (*)$$

Proof.

Choose the bits of a sequence $S \in \{0,1\}^\omega$ by independent tosses of a fair coin.

Kraft's Inequality

If $A \subseteq \{0,1\}^*$ is prefix-free, then

$$\sum_{x \in A} 2^{-|x|} \leq 1. \qquad (*)$$

Proof.

Choose the bits of a sequence $S \in \{0,1\}^\omega$ by independent tosses of a fair coin.
The left-hand side of $(*)$ is the probability that S has a prefix in A. $\square$

Kolmogorov Complexity

Definition

The algorithmic a priori probability of a string $x \in \{0,1\}^*$ is

$$\boldsymbol{m}(x) = \sum_{\substack{\pi \in \{0,1\}^* \\ U(\pi)=x}} 2^{-|\pi|}. \quad \text{(random programming!)}$$

Kolmogorov Complexity

Definition

The algorithmic a priori probability of a string $x \in \{0,1\}^*$ is

$$\boldsymbol{m}(x) = \sum_{\substack{\pi \in \{0,1\}^* \\ U(\pi)=x}} 2^{-|\pi|}. \quad \text{(random programming!)}$$

By Kraft's inequality, $\boldsymbol{m}$ is a subprobability measure on $\{0,1\}^*$.

Kolmogorov Complexity

Definition

The algorithmic a priori probability of a string $x \in \{0,1\}^*$ is

$$\boldsymbol{m}(x) = \sum_{\substack{\pi \in \{0,1\}^* \\ U(\pi)=x}} 2^{-|\pi|}. \quad \text{(random programming!)}$$

By Kraft's inequality, $\boldsymbol{m}$ is a subprobability measure on $\{0,1\}^*$.

Trivially, $2^{-K(x)} \leq \boldsymbol{m}(x)$.

Kolmogorov Complexity

Definition

The algorithmic a priori probability of a string $x \in \{0,1\}^*$ is

$$\boldsymbol{m}(x) = \sum_{\substack{\pi \in \{0,1\}^* \\ U(\pi)=x}} 2^{-|\pi|}. \quad \text{(random programming!)}$$

By Kraft's inequality, $\boldsymbol{m}$ is a subprobability measure on $\{0,1\}^*$.

Trivially, $2^{-K(x)} \leq \boldsymbol{m}(x)$.

Levin's Coding Theorem (1974)

There is a constant $\alpha > 0$ such that, for all $x \in \{0,1\}^*$,

$$\boldsymbol{m}(x) \leq \alpha 2^{-K(x)}.$$

Kolmogorov Complexity

Definition

The algorithmic a priori probability of a string $x \in \{0,1\}^*$ is

$$\boldsymbol{m}(x) = \sum_{\substack{\pi \in \{0,1\}^* \\ U(\pi)=x}} 2^{-|\pi|}. \quad \text{(random programming!)}$$

By Kraft's inequality, $\boldsymbol{m}$ is a subprobability measure on $\{0,1\}^*$.

Trivially, $2^{-K(x)} \leq \boldsymbol{m}(x)$.

Levin's Coding Theorem (1974)

There is a constant $\alpha > 0$ such that, for all $x \in \{0,1\}^*$,

$$\boldsymbol{m}(x) \leq \alpha 2^{-K(x)}.$$

"x cannot have many long programs without having a short program."

Kolmogorov Complexity

We just saw that there is a constant $\alpha > 0$ such that, for all strings $x \in \{0,1\}^*$,

$$2^{-K(x)} \leq \boldsymbol{m}(x) \leq \alpha 2^{-K(x)}.$$

Kolmogorov Complexity

We just saw that there is a constant $\alpha > 0$ such that, for all strings $x \in \{0,1\}^*$,

$$2^{-K(x)} \leq \boldsymbol{m}(x) \leq \alpha 2^{-K(x)}.$$

Taking logarithms, we thus have a constant $c \in \mathbb{N}$ such that, for all $x \in \{0,1\}^*$,

$$\left| K(x) - \log \frac{1}{\boldsymbol{m}(x)} \right| \leq c.$$

Kolmogorov Complexity

We just saw that there is a constant $\alpha > 0$ such that, for all strings $x \in \{0,1\}^*$,

$$2^{-K(x)} \leq \boldsymbol{m}(x) \leq \alpha 2^{-K(x)}.$$

Taking logarithms, we thus have a constant $c \in \mathbb{N}$ such that, for all $x \in \{0,1\}^*$,

$$\left| K(x) - \log \frac{1}{\boldsymbol{m}(x)} \right| \leq c.$$

This fact, due to Levin (1974) says that, to within an additive constant,

Kolmogorov complexity is Shannon self-information with respect to $\boldsymbol{m}$!

Dimensions of Finite Strings

Our next task: Extend algorithmic dimension to define $\dim(x)$ for each $x \in \{0,1\}^*$.

Dimensions of Finite Strings

Our next task: Extend algorithmic dimension to define $\dim(x)$ for each $x \in \{0,1\}^*$.

Notation:

$$\mathcal{T} = \{0,1\}^* \cup \{0,1\}^*\square$$

Dimensions of Finite Strings

Our next task: Extend algorithmic dimension to define $\dim(x)$ for each $x \in \{0,1\}^*$.

Notation:

$$\mathcal{T} = \{0,1\}^* \cup \underbrace{\{0,1\}^*\square}_{\substack{\text{terminated} \\ \text{binary strings}}}$$

Dimensions of Finite Strings

Our next task: Extend algorithmic dimension to define $\dim(x)$ for each $x \in \{0,1\}^*$.

Notation:

$$\mathcal{T} = \underbrace{\{0,1\}^*}_{\substack{\text{prefixes}\\ \text{thereof}}} \cup \underbrace{\{0,1\}^*\square}_{\substack{\text{terminated}\\ \text{binary strings}}}$$

Dimensions of Finite Strings

Our next task: Extend algorithmic dimension to define $\dim(x)$ for each $x \in \{0,1\}^*$.

Notation:

$$\mathcal{T} = \underbrace{\{0,1\}^*}_{\substack{\text{prefixes}\\ \text{thereof}}} \cup \underbrace{\{0,1\}^*\square}_{\substack{\text{terminated}\\ \text{binary strings}}}$$

Definition

For $s \in [0,\infty)$, an $s-$termgale is a function $d : \mathcal{T} \to [0,\infty)$ satisfying

$$d(\lambda) \leq 1$$

and

$$d(w) = 2^{-s}[d(w0) + d(w1) + d(w\square)]$$

for all $x \in \{0,1\}^*$.

Dimensions of Finite Strings

d bets on the successive bits and on the termination of a finite string.

Dimensions of Finite Strings

d bets on the successive bits and on the termination of a finite string.

Example

Define $d : \mathcal{T} \to [0, \infty)$ by $d(\lambda) = 1$,

$$d(w0) = \frac{3}{2} d(w),$$
$$d(w1) = d(w\square) = \frac{1}{4} d(w).$$

This is a 1-termgale.

Dimensions of Finite Strings

d bets on the successive bits and on the termination of a finite string.

Example

Define $d : \mathcal{T} \to [0, \infty)$ by $d(\lambda) = 1$,

$$d(w0) = \frac{3}{2} d(w),$$
$$d(w1) = d(w\square) = \frac{1}{4} d(w).$$

This is a 1-termgale.
If $w \in \{0,1\}^n$ has n_0 $0s$ and n_1 $1s$, then

$$d(w\square) = \left(\frac{3}{2}\right)^{n_0} \left(\frac{1}{4}\right)^{n_1+1} = 2^{n_0(1+\log 3)-2(n+1)}.$$

Dimensions of Finite Strings

d bets on the successive bits and on the termination of a finite string.

Example

Define $d : \mathcal{T} \to [0, \infty)$ by $d(\lambda) = 1$,

$$d(w0) = \frac{3}{2} d(w),$$
$$d(w1) = d(w\square) = \frac{1}{4} d(w).$$

This is a 1-termgale.
If $w \in \{0,1\}^n$ has n_0 $0s$ and n_1 $1s$, then

$$d(w\square) = \left(\frac{3}{2}\right)^{n_0} \left(\frac{1}{4}\right)^{n_1+1} = 2^{n_0(1+\log 3)-2(n+1)}.$$

$\therefore$ If $n_0 >> 0.77(n+1)$, then $d(w\square) >> d(\lambda)$, even though d loses $\frac{3}{4}$ of its money when $\square$ appears.

Dimensions of Finite Strings

Trivial Observation

Dimensions of Finite Strings

Trivial Observation

If

$$2^{-s|x|}d(x) = 2^{-s'|x|}d'(x) \qquad (\star)$$

for all $x \in \mathcal{T}$, then d is an $s-$termgale iff d' is an s'-termgale.

Dimensions of Finite Strings

Trivial Observation

If

$$2^{-s|x|}d(x) = 2^{-s'|x|}d'(x) \qquad (\star)$$

for all $x \in \mathcal{T}$, then d is an $s-$termgale iff d' is an s'-termgale.
Hence, if d is a $0-$termgale, then

$$d'(x) = 2^{s|x|}d(x)$$

is an $s-$termgale, and all $s-$termgales are of this form.

Dimensions of Finite Strings

Definition

A termgale is a family

$$d = \{d^{(s)} | s \in [0, \infty)\}$$

of $s-$termgales, one for each s, related by $(\star)$.

Dimensions of Finite Strings

Definition

A termgale is a family

$$d = \{d^{(s)} | s \in [0, \infty)\}$$

of $s-$termgales, one for each s, related by $(\star)$.

d is completely determined by any one of its elements.

Dimensions of Finite Strings

Definition

A termgale is a family

$$d = \{d^{(s)} | s \in [0, \infty)\}$$

of $s-$termgales, one for each s, related by $(\star)$.

d is completely determined by any one of its elements.

Definition

Let d be a termgale, $a \in \mathbb{Z}^+$, and $w \in \{0,1\}^*$. The dimension of w relative to d at significance level a is

$$\dim_d^a(w) = \inf\{s | d^{(s)}(w\square) > a\}.$$

Dimensions of Finite Strings

Definition

A termgale is a family

$$d = \{d^{(s)} | s \in [0, \infty)\}$$

of $s-$termgales, one for each s, related by $(\star)$.

d is completely determined by any one of its elements.

Definition

Let d be a termgale, $a \in \mathbb{Z}^+$, and $w \in \{0,1\}^*$. The dimension of w relative to d at significance level a is

$$\dim_d^a(w) = \inf\{s | d^{(s)}(w\square) > a\}.$$

We write $\dim_d(w) = \dim_d^1(w)$.

Dimensions of Finite Strings

We have now discretized Hausdorff dimension.

Dimensions of Finite Strings

We have now discretized Hausdorff dimension.

Definition

A termgale d is algorithmic if $d^{(0)}$ is lower semicomputable.

Dimensions of Finite Strings

We have now discretized Hausdorff dimension.

Definition

A termgale d is algorithmic if $d^{(0)}$ is lower semicomputable.

Now optimize:

Definition

An algorithmic termgale $\tilde{d}$ is optimal if, for every algorithmic termgale d, there is a constant $\alpha > 0$ such that, for all $s \in [0, \infty)$ and $w \in \{0,1\}^*$,

$$\tilde{d}^{(s)}(w\square) \geq \alpha d^{(s)}(w\square).$$

Dimensions of Finite Strings

Theorem (J. Lutz 2003)

If $\tilde{d}$ is an optimal algorithmic termgale, then, for every algorithmic termgale d and every $a \in \mathbb{Z}^+$, there is a constant $\nu \in [0, \infty)$ such that, for all $w \in \{0,1\}^*$,

$$\dim^a_{\tilde{d}}(w) \leq \dim_d(w) + \frac{\nu}{1 + |w|}.$$

Dimensions of Finite Strings

Theorem (J. Lutz 2003)

If $\tilde{d}$ is an optimal algorithmic termgale, then, for every algorithmic termgale d and every $a \in \mathbb{Z}^+$, there is a constant $\nu \in [0, \infty)$ such that, for all $w \in \{0,1\}^*$,

$$\dim_{\tilde{d}}^{a}(w) \leq \dim_d(w) + \frac{\nu}{1 + |w|}.$$

Corollary

If d_1 and d_2 are optimal algorithmic termgales and $a_1, a_2 \in \mathbb{Z}^+$, then there is a constant $\alpha \in [0, \infty)$ such that, for all $w \in \{0,1\}^*$,

$$\left|\dim_{d_1}^{a_1}(w) - \dim_{d_2}^{a_2}(w)\right| \leq \frac{\alpha}{1 + |w|}.$$

Dimensions of Finite Strings

Theorem (J. Lutz 2003)

If $\tilde{d}$ is an optimal algorithmic termgale, then, for every algorithmic termgale d and every $a \in \mathbb{Z}^+$, there is a constant $\nu \in [0, \infty)$ such that, for all $w \in \{0,1\}^*$,

$$\dim_{\tilde{d}}^{a}(w) \leq \dim_d(w) + \frac{\nu}{1 + |w|}.$$

Corollary

If d_1 and d_2 are optimal algorithmic termgales and $a_1, a_2 \in \mathbb{Z}^+$, then there is a constant $\alpha \in [0, \infty)$ such that, for all $w \in \{0,1\}^*$,

$$\left|\dim_{d_1}^{a_1}(w) - \dim_{d_2}^{a_2}(w)\right| \leq \frac{\alpha}{1 + |w|}.$$

Hence it makes very little difference which optimal algorithmic termgale or which significance level we use.

Dimensions of Finite Strings

Theorem (J. Lutz 2003)

There is an optimal algorithmic termgale $\boldsymbol{d}_{\square}$.

The proof uses Levin's $\boldsymbol{m}$.

Dimensions of Finite Strings

Theorem (J. Lutz 2003)

There is an optimal algorithmic termgale $\boldsymbol{d}_{\square}$.

The proof uses Levin's $\boldsymbol{m}$.

Definition

The dimension of a string $w \in \{0,1\}^*$ is

$$\dim(w) = \dim^1_{\boldsymbol{d}_{\square}}(w).$$

Dimensions of Finite Strings

Theorem (J. Lutz 2003)

There is an optimal algorithmic termgale $\boldsymbol{d}_{\square}$.

The proof uses Levin's $\boldsymbol{m}$.

Definition

The dimension of a string $w \in \{0,1\}^*$ is

$$\dim(w) = \dim^1_{\boldsymbol{d}_{\square}}(w).$$

Theorem (J. Lutz 2003)

There is a constant $c \in \mathbb{N}$ such that, for all $x \in \{0,1\}^*$,

$$|K(x) - |x| \dim(x)| \leq c.$$

Dimensions of Finite Strings

Theorem (J. Lutz 2003)

There is an optimal algorithmic termgale $\boldsymbol{d}_\square$.

The proof uses Levin's $\boldsymbol{m}$.

Definition

The dimension of a string $w \in \{0,1\}^*$ is

$$\dim(w) = \dim^1_{\boldsymbol{d}_\square}(w).$$

Theorem (J. Lutz 2003)

There is a constant $c \in \mathbb{N}$ such that, for all $x \in \{0,1\}^*$,

$$|K(x) - |x| \dim(x)| \leq c.$$

$\therefore$ $\dim(x)$ is the density of algorithmic information in x.

Dimensions of Finite Strings

Summary:

Up to constant additive terms,

$$K(x) = \log \frac{1}{\boldsymbol{m}(x)} = |x| \dim(x).$$

Dimensions of Finite Strings

Summary:

Up to constant additive terms,

$$K(x) = \log \frac{1}{\boldsymbol{m}(x)} = |x| \dim(x).$$

The great, and fundamentally different, theories of Hausdorff (1919), Shannon (1948), and Kolmogorov (1965) are in exquisite agreement on the information in finite strings.

Thank you!

Thanks to CBMS and to NSF grants 2329555 and 1900716.