Algorithmic Randomness

Nan Fang

Department of Philosophy, Peking University

2014.9.16

What is randomness

An example of random sequence by tossing coin:
010100110111101101110101000000101111...

What is randomness

An example of random sequence by tossing coin:
0101001101111011011101010100000101111...

Typicalness: (The statistician's approach)A random object is the typical outcome of a random variable. Random sequences should not have effectively rare properties.

What is randomness

An example of random sequence by tossing coin:
0101001101111011011101010100000101111...

Typicalness: (The statistician's approach)A random object is the typical outcome of a random variable. Random sequences should not have effectively rare properties.

Unpredictability: (The gambler's approach)A random object should be impossible to predict.

What is randomness

An example of random sequence by tossing coin:
0101001101111011011101010100000101111...

Typicalness: (The statistician's approach)A random object is the typical outcome of a random variable. Random sequences should not have effectively rare properties.

Unpredictability: (The gambler's approach)A random object should be impossible to predict.

Incompressibility: (The coder's approach)A random object should not have a shorter description than itself. No effective martingale (betting) can make an infinite amount betting of the bits.

Notations

We will study randomness for infinite binary sequences.

Notations

We will study randomness for infinite binary sequences.

- We represent natural number x by the finite string, the binary representation of $x + 1$ with the first bit 1 removed.

Notations

We will study randomness for infinite binary sequences.

- We represent natural number x by the finite string, the binary representation of $x + 1$ with the first bit 1 removed.
- We interpret sequences $X \in 2^{\mathbb{N}}$ as sets of natural numbers, $S_X = \{n \in \mathbb{N} : X(n) = 1\}$, or as real numbers in $[0, 1]$, $\alpha_X = \sum_n X(n)2^{-n}$.

the statisticians approach

- von Mises, 1919 : A random sequence should have as many 0's as 1's. But what about 10101010101010101010......

the statisticians approach

- von Mises, 1919 : A random sequence should have as many 0's as 1's. But what about 1010101010101010......
- von Mises idea: For some particular increasing functions on $\mathbb{N}$, if we select a subsequence $\{a_{f(0)}, a_{f(1)}, a_{f(2)}, ...\}$ then the number of 0's and 1's divided by the number of elements selected should end to $\frac{1}{2}$. (Law of Large Numbers)

the statisticians approach

- von Mises, 1919 : A random sequence should have as many 0's as 1's. But what about 10101010101010......
- von Mises idea: For some particular increasing functions on $\mathbb{N}$, if we select a subsequence $\{a_{f(0)}, a_{f(1)}, a_{f(2)}, ...\}$ then the number of 0's and 1's divided by the number of elements selected should end to $\frac{1}{2}$. (Law of Large Numbers)
- Church: We should just allow all computable functions.

the statisticians approach

- von Mises, 1919 : A random sequence should have as many 0's as 1's. But what about 10101010101010......
- von Mises idea: For some particular increasing functions on $\mathbb{N}$, if we select a subsequence $\{a_{f(0)}, a_{f(1)}, a_{f(2)}, ...\}$ then the number of 0's and 1's divided by the number of elements selected should end to $\frac{1}{2}$. (Law of Large Numbers)
- Church: We should just allow all computable functions.
- Ville, 1939 : No countable selection possible!

the statisticians approach

- von Mises, 1919 : A random sequence should have as many 0's as 1's. But what about 101010101010101010......
- von Mises idea: For some particular increasing functions on $\mathbb{N}$, if we select a subsequence $\{a_{f(0)}, a_{f(1)}, a_{f(2)}, ...\}$ then the number of 0's and 1's divided by the number of elements selected should end to $\frac{1}{2}$. (Law of Large Numbers)
- Church: We should just allow all computable functions.
- Ville, 1939 : No countable selection possible!
- Martin-Löf, 1966: Using shrinking effective null sets as representing effective tests.

Martin-Löf Tests and Randomness

Definition

- A Martin-Löf (ML) test (for Lebesgue measure) is a recursively enumerable set $W \subset \mathbb{N} \times 2^{<\mathbb{N}}$ such that, if we let $W_n = \{\sigma : (n, \sigma) \in W\}$, for all $n \in \mathbb{N}$,

$$\sum_{\sigma \in W_n} 2^{-|\sigma|} < 2^{-n}.$$

Martin-Löf Tests and Randomness

Definition

- A Martin-Löf (ML) test (for Lebesgue measure) is a recursively enumerable set $W \subset \mathbb{N} \times 2^{<\mathbb{N}}$ such that, if we let $W_n = \{\sigma : (n, \sigma) \in W\}$, for all $n \in \mathbb{N}$,

$$\sum_{\sigma \in W_n} 2^{-|\sigma|} < 2^{-n}.$$

- A sequence $X \in 2^{\mathbb{N}}$ fails the test if $X \in \bigcap_m G_m$, otherwise X passes the test.

Martin-Löf Tests and Randomness

Definition

- A Martin-Löf (ML) test (for Lebesgue measure) is a recursively enumerable set $W \subset \mathbb{N} \times 2^{<\mathbb{N}}$ such that, if we let $W_n = \{\sigma : (n, \sigma) \in W\}$, for all $n \in \mathbb{N}$,

$$\sum_{\sigma \in W_n} 2^{-|\sigma|} < 2^{-n}.$$

- A sequence $X \in 2^{\mathbb{N}}$ fails the test if $X \in \bigcap_m G_m$, otherwise X passes the test.
- X is Martin-Löf (ML) random if X passes each ML-test.

Universal ML-test

Proposition

The class of all ML-random sequences is conull.

Universal ML-test

Proposition

The class of all ML-random sequences is conull.

Proposition

There exist a universal ML-test U such that X is ML-random iff X is not covered by $\bigcap_n U_n$.

Universal ML-test

Proposition

The class of all ML-random sequences is conull.

Proposition

There exist a universal ML-test U such that X is ML-random iff X is not covered by $\bigcap_n U_n$.

- Enumerate all c.e. sets $W^{(e)} \subset \mathbb{N} \times 2^{<\mathbb{N}}$, stopping should one violated the measure condition of some $W_n^{(e)}$.

Universal ML-test

Proposition

The class of all ML-random sequences is conull.

Proposition

There exist a universal ML-test U such that X is ML-random iff X is not covered by $\bigcap_n U_n$.

- Enumerate all c.e. sets $W^{(e)} \subset \mathbb{N} \times 2^{<\mathbb{N}}$, stopping should one violated the measure condition of some $W_n^{(e)}$.
- Then we can define a universal test U by letting

$$U_n = \bigcup_e W_{n+e+1}^{(e)}$$

Martingales

Definition

A betting strategy b is a function $b : 2^{<\mathbb{N}} \to \{0,1\} \times [0,1]$.

Martingales

Definition

A betting strategy b is a function $b : 2^{<\mathbb{N}} \to \{0,1\} \times [0,1]$.

We can keep track of the player's capital through a function:

Martingales

Definition

A betting strategy b is a function $b : 2^{<\mathbb{N}} \to \{0,1\} \times [0,1]$.

We can keep track of the player's capital through a function:

Definition

A martingale is a function $F : 2^{<\mathbb{N}} \to [0,\infty)$ satisfies

$$F(\sigma 0) + F(\sigma 1) = 2F(\sigma), \text{ for every } \sigma \in 2^{<\mathbb{N}}$$

Martingales

Definition

A betting strategy b is a function $b : 2^{<\mathbb{N}} \to \{0,1\} \times [0,1]$.

We can keep track of the player's capital through a function:

Definition

A martingale is a function $F : 2^{<\mathbb{N}} \to [0,\infty)$ satisfies

$$F(\sigma 0) + F(\sigma 1) = 2F(\sigma), \text{ for every } \sigma \in 2^{<\mathbb{N}}$$

A martingale F is successful on an infinite sequence X if

$$\limsup_{n\to\infty} F(X \upharpoonright_n) = \infty$$

c.e. martingales and ML randomness

Definition

A function $F : 2^{<\mathbb{N}} \to \mathbb{R}$ is computably enumerable(c.e.) if there exists, uniformly in σ, a recursive nondecreasing sequence $(q_k^{(\sigma)})$ of rational numbers such that $q_k^{(\sigma)} \to F(\sigma)$, or equivalently, the left cut of $F(\sigma)$ is uniformly enumerable, i.e. the set $\{(q, \sigma) : q < F(\sigma)\}$ is c.e.

c.e. martingales and ML randomness

Definition

A function $F : 2^{<\mathbb{N}} \to \mathbb{R}$ is computably enumerable(c.e.) if there exists, uniformly in σ, a recursive nondecreasing sequence $(q_k^{(\sigma)})$ of rational numbers such that $q_k^{(\sigma)} \to F(\sigma)$,
or equivalently, the left cut of $F(\sigma)$ is uniformly enumerable, i.e. the set $\{(q, \sigma) : q < F(\sigma)\}$ is c.e.

Theorem

A sequence X is ML-random if and only if no c.e. martingale succeeds on it.

Machine complexity

- Let M be a Turing machine. M computes a partial recursive function $2^{<\mathbb{N}} \to 2^{<\mathbb{N}}$.

Machine complexity

- Let M be a Turing machine. M computes a partial recursive function $2^{<\mathbb{N}} \to 2^{<\mathbb{N}}$.
- We define the M-complexity of a string x as

$$C_M(x) = \min\{|\sigma| : M(\sigma) = x\}$$

where $\min \emptyset = \infty$.

Machine complexity

- Let M be a Turing machine. M computes a partial recursive function $2^{<\mathbb{N}} \to 2^{<\mathbb{N}}$.
- We define the M-complexity of a string x as

$$C_M(x) = \min\{|\sigma| : M(\sigma) = x\}$$

where $\min \emptyset = \infty$.

- A machine R is optimal if for every machine M there exists a constant e_M such that

$$(\forall x)[C_R(x) \leq C_M(x) + e_M]$$

.

The Invariance Theorem

Theorem (Kolmogorov)

There exists an optimal machine R.

The Invariance Theorem

Theorem (Kolmogorov)

There exists an optimal machine R.

- Let (M_e) be an effective enumeration of all Turing machines.

The Invariance Theorem

Theorem (Kolmogorov)

There exists an optimal machine R.

- Let (M_e) be an effective enumeration of all Turing machines.
- On input σ, R parses σ and finds unique e and τ such that $\sigma = 0^e 1\tau$. Then let R outputs $M_e(\tau)$.

The Invariance Theorem

Theorem (Kolmogorov)

There exists an optimal machine R.

- Let (M_e) be an effective enumeration of all Turing machines.
- On input σ, R parses σ and finds unique e and τ such that $\sigma = 0^e 1\tau$. Then let R outputs $M_e(\tau)$.
- Then we have $R(0^e 1\tau) = M_e(\tau)$, and $(\forall x)[C_R(x) \leq C_{M_e}(x) + e + 1]$.

Kolmogorov complexity

- For two function f and g, if there exist a constant c such that for all x, $f(x) \leq g(x) + c$, we write $f \leq^+ g$.
- $f =^+ g$ if $f \leq^+ g$ and $g \leq^+ f$
- For any two optimal machine R and S, we have $C_R =^+ C_S$.

Kolmogorov complexity

- For two function f and g, if there exist a constant c such that for all x, $f(x) \leq g(x) + c$, we write $f \leq^+ g$.
- $f =^+ g$ if $f \leq^+ g$ and $g \leq^+ f$
- For any two optimal machine R and S, we have $C_R =^+ C_S$.
- We define the (plain) Kolmogorov complexity of a string x as

$$C(x) = C_R(x)$$

Properties of C

- There exists an e such that for all x, $C(x) \leq |x| + e$.

Properties of C

- There exists an e such that for all x, $C(x) \leq |x| + e$. Actually, e is the index of the copying machine.

Properties of C

- There exists an e such that for all x, $C(x) \leq |x| + e$. Actually, e is the index of the copying machine.
- For each length n, there exist incompressible strings of length n, i.e. strings x with $C(x) \geq |x|$.

Properties of C

- There exists an e such that for all x, $C(x) \leq |x| + e$.
  Actually, e is the index of the copying machine.
- For each length n, there exist incompressible strings of length n, i.e. strings x with $C(x) \geq |x|$.
  Because there are only $\sum_{k=0}^{n-1} 2^k = 2^n - 1$ programs of length $< n$.

Properties of C

- There exists an e such that for all x, $C(x) \leq |x| + e$. Actually, e is the index of the copying machine.
- For each length n, there exist incompressible strings of length n, i.e. strings x with $C(x) \geq |x|$.
  Because there are only $\sum_{k=0}^{n-1} 2^k = 2^n - 1$ programs of length $< n$.
- Then we can see that $\forall x[C(x) \leq^+ |x|]$ and $\exists^\infty x[C(x) \geq^+ |x|]$, we say that $|x|$ is an infinitely often tight upper bound of $C(x)$.

Weakness of C

Theorem (Martin-Löf)

Let $k \in \mathbb{N}$. For any sufficiently long string x there exists an initial segment $y \preceq x$ such that $C(y) < |y| - k$.

Weakness of C

Theorem (Martin-Löf)

Let $k \in \mathbb{N}$. For any sufficiently long string x there exists an initial segment $y \preceq x$ such that $C(y) < |y| - k$.

Corollary

Let $k \in \mathbb{N}$. There exists an x such that for some splitting $x = yz$ we have $C(x) > C(y) + C(z) + k$.

Prefix-free machine

Definition

A set $W \supset 2^{<\mathbb{N}}$ is prefix-free if for any $x, y \in W$,

$$x \preceq y \implies x = y$$

.

Prefix-free machine

Definition

A set $W \supset 2^{<\mathbb{N}}$ is prefix-free if for any $x, y \in W$,

$$x \preceq y \implies x = y$$

.

A machine M is prefix-free if its domain is a prefix-free set.

prefix-free Kolmogorov complexity

Similarly,
A prefix-free machine S is optimal if for every prefix-free machine M there exists a constant e_M such that

$$(\forall x)[C_S(x) \leq C_M(x) + e_M]$$

prefix-free Kolmogorov complexity

Similarly,
A prefix-free machine S is optimal if for every prefix-free machine M there exists a constant e_M such that

$$(\forall x)[C_S(x) \leq C_M(x) + e_M]$$

Definition

The prefix-free complexity of a string x is defined as

$$K(x) = C_S(x)$$

existence of optimal prefix-free machine

Proposition

There exists an optimal prefix-free machine S.

existence of optimal prefix-free machine

Proposition

There exists an optimal prefix-free machine S.

Proof.

- Enumerate all Turing machine.

existence of optimal prefix-free machine

Proposition

There exists an optimal prefix-free machine S.

Proof.

- Enumerate all Turing machine.
- Whenever we see that some machin M_e is not prefix-free, we stop enumerating its domain. This way we convert it to a prefix-free machine $\tilde{M}_e$. If M_e is already prefix-free, it remains unaltered.

existence of optimal prefix-free machine

Proposition

There exists an optimal prefix-free machine S.

Proof.

- Enumerate all Turing machine.
- Whenever we see that some machin M_e is not prefix-free, we stop enumerating its domain. This way we convert it to a prefix-free machine $\tilde{M}_e$. If M_e is already prefix-free, it remains unaltered.
- Then $(\tilde{M}_e)$ is an enumeration of all prefix-free machine, we define $S(0^e 1\sigma) = \tilde{M}_e(\sigma)$.

□

Properties of K

Consider the upper bounds of K:

Properties of K

Consider the upper bounds of K:

- The copying machine is not prefix-free, but the machine $M(0^{|x|}1x) = x$ is prefix-free. So we have

$$K(x) \leq^{+} 2|x|$$

.

Properties of K

Consider the upper bounds of K:

- The copying machine is not prefix-free, but the machine $M(0^{|x|}1x) = x$ is prefix-free. So we have

$$K(x) \leq^{+} 2|x|$$

  .
- Actually, we can get $K(x) \leq^{+} |x| + K(|x|) \leq^{+} |x| + 2\log|x|$.

Schnorr Theorem

Theorem (Schnorr)

A sequence X is ML-random iff there exists a c such that for all n,

$$K(X \upharpoonright_n) \geq n - c$$

Halting probability

The halting probability of a prefix-free machine M is

$$\Omega_M = \sum_{\sigma \in \mathrm{dom}(M)} 2^{-|\sigma|}$$

Halting probability

The halting probability of a prefix-free machine M is

$$\Omega_M = \sum_{\sigma \in \mathrm{dom}(M)} 2^{-|\sigma|}$$

Let $\Omega = \Omega_S$

Theorem (Chaitin)

Ω is ML-random.

Other randomness notions

weaker randomness notions:

- Schnorr randomness
- Computable randomness
- Resource-bounded randomness

Other randomness notions

weaker randomness notions:

- Schnorr randomness
- Computable randomness
- Resource-bounded randomness

stronger randomness notions:

- weak-2-randomness
- 2-randomness, n-randomness

formalizing the notion of computability

- Gödel, 1933: general recursive functions.
- Church, 1936: λ-calculus.
- Turing, 1936: Turing Machine.

formalizing the notion of computability

- Gödel, 1933: general recursive functions.
- Church, 1936: λ-calculus.
- Turing, 1936: Turing Machine.

Church-Turing Thesis

A function on the natural numbers is computable in an informal sense (i.e., computable by a human being using a pencil-and-paper method, ignoring resource limitations) if and only if it is computable by a Turing machine.

Martin-Löf-Chaitin Thesis

Martin-Löf-Chaitin Thesis

Martin-Löf randomness captures our commonly held intuitions about randomness.

Martin-Löf-Chaitin Thesis

Martin-Löf-Chaitin Thesis

Martin-Löf randomness captures our commonly held intuitions about randomness.

As Gödel noted, with the definition of computability “one has for the first time succeeded in giving an absolute definition of an interesting epistemological notion, i.e., one not depending on the formalism chosen” (Collected Works, Volume II, p.150).

Martin-Löf-Chaitin Thesis

Martin-Löf-Chaitin Thesis

Martin-Löf randomness captures our commonly held intuitions about randomness.

As Gödel noted, with the definition of computability "one has for the first time succeeded in giving an absolute definition of an interesting epistemological notion, i.e., one not depending on the formalism chosen" (Collected Works, Volume II, p.150). However, for randomness we cannot get this absoluteness in Gödel's sense.

Martin-Löf-Chaitin Thesis

Martin-Löf-Chaitin Thesis

Martin-Löf randomness captures our commonly held intuitions about randomness.

As Gödel noted, with the definition of computability “one has for the first time succeeded in giving an absolute definition of an interesting epistemological notion, i.e., one not depending on the formalism chosen” (Collected Works, Volume II, p.150).
However, for randomness we cannot get this absoluteness in Gödel’s sense.
Hence comes the so-called "The No-Thesis Thesis".

References

- Nies, Andre. *Computability and Randomness*. Oxford: Oxford UP, 2009.
- Downey, Ronald G., and Denis R. Hirschfeldt. *Algorithmic Randomness and Complexity*. Springer, 2010.
- Christopher P. Porter *Mathematical and Philosophical Perspectives on Algorithmic Randomness*. Doctoral Dissertation, 2012.