Chaos, Complexity, and Inference (36-462)

Lecture 8

Cosma Shalizi

7 February 2008

- In what senses can we say that chaos gives us deterministic randomness?
- Explaining “random” in terms of information
- Chaotic dynamics and information

All ideas shamelessly stolen from [1]
Single most important reference on algorithmic definition of randomness: [2]
But see also [3] on detailed connections to dynamics

Probability Theory and Its Models

Probability theory is a **theory** — axioms & logical consequences

Something which obeys that theory is one of its **realizations**

E.g., $r = 1$ logistic map, with usual generating partition, is a realization of the theory of IID fair coin tossing

Can we say something general about realizations of probability theory?

Compression

Information theory last time: looked at compact coding random objects
Coding and compression turn out to define randomness
Lossless compression: Encoded version is shorter than original, but can uniquely & exactly recover original
Lossy compression: Can only get something *close* to original
Stick with lossless compression

Lossless compression needs an **effective procedure** — definite steps which a machine could take to recover the original

Effective procedures are the same as algorithms

Algorithms are the same as recursive functions

Recursive functions are the same as what you can do with a finite state machine and an unlimited external memory (Turing machine)

For concreteness, think about programs written in a universal programming language (Lisp, Fortran, C, C++, Pascal, Java, Perl, OCaml, Forth, ...)

x is our object, size $|x|$

Desired: a program in language L which will output x and then stop

some trivial programs *eventually* output everything

e.g. 01234567891011121314...

those programs are descriptions of x

What is the *shortest* program which will do this?

N.B.: `print(x);` is the *upper bound* on the description length

finite # programs shorter than that

so there must be a shortest

Length of this shortest program is $K_L(x)$

Why the big deal about universal computer?

1. Want to handle as general a situation as possible
2. Emulation: for any other universal language M, can write a compiler or translator from L to M, so

$$K_M(x) \leq |C_{L \rightarrow M}| + K_L(x)$$

Which universal language doesn't matter, much; and could use any other model of computation

Kolmogorov Complexity

The **Kolmogorov complexity** of x, relative to L, is

$$K_L(x) = \min_{p \in \mathcal{D}(x)} |p|$$

where $\mathcal{D}(x) =$ all programs in L that output x and then halt
This is the **algorithmic information content** of x
a.k.a. Kolmogorov-Chaitin complexity,
Kolmogorov-Chaitin-Solomonoff complexity...

$$1 \leq K_L(x) \leq |x| + c$$

where c is the length of the “print this” stuff
If $K_L(x) \approx |x|$, then x is **incompressible**

Examples

"0": $K \leq 1 + c$
"0" ten thousand times: $K \leq 1 + \log_2 10^4 + c = 1 + 4\log_2 10 + c$
"0" ten billion times: $K \leq 1 + 10\log_2 10 + c$
"10010010" ten billion times: $K \leq 8 + 10\log_2 10 + c$
π, first n digits: $K \leq \gamma + \log_2 n$
In fact, any number you care to name contains little algorithmic information

Why?

Incompressibility and randomness

Most objects are not very compressible
Exactly 2^n objects of length n bits
At most 2^k programs of length k bits
No more than 2^k n-bit objects can be compressed to k bits
Proportion is $\leq 2^{k-n}$
At most $2^{-n/2}$ objects can be compressed in half
Vast majority of sequences from a uniform IID source will be incompressible
"uniform IID" = "pure noise" for short

More compressibility and randomness

Suppose x is a binary string of length n, with $n \gg 1$
If proportion of 1s in x is p, then

$$K(x) \leq -n(p \log_2 p + (1-p) \log_2 1 - p) + o(n) = nH(p) + o(n)$$

$nH(p) < n$ if $p \neq \frac{1}{2}$
Similarly for statistics of pairs, triples, ...
Suggests:

1. Most sequences from non-pure-noise sources will be compressible
2. Incompressible sequences look like pure noise

ANY SIGNAL DISTINGUISHABLE FROM NOISE IS INSUFFICIENTLY COMPRESSED

Incompressible sequences look random

CLAIM 1: Incompressible sequences have all the *effectively testable* properties of pure noise
CLAIM 2: Sequences which fail to have the testable properties of pure noise are compressible
Redundancy $|x| - K_L(x)$ is distance from pure noise
If X *is* pure noise,

$$\Pr\left(|X| - K_L(X) > c\right) \leq 2^{-c}$$

Power of this test is close to that of any other (computable) test

Why the L doesn't matter

Take your favorite sequence x
In new language L', the program "!" produces x, any program not beginning "!" is in L
Can make $K_{L'}(x) = 1$, but makes others longer
But the trick doesn't keep working
can translate between languages with constant complexity
still true that large incompressible sequences look like pure noise

ANY DETERMINISM DISTINGUISHABLE FROM RANDOMNESS IS INSUFFICIENTLY COMPLEX

Poincaré said as much 100 years ago, without the math [4]
Excerpt on website
Extends to other, partially-compressible stochastic processes
The maximally-compressed description is incompressible
so other stochastic processes are transformations of noise

The Problem

There is no algorithm to compute $K_L(x)$
Suppose there was such a program, U for universal
Use it to make a new program C

1. Sort all sequences by length and then alphabetically
2. For sequence x_t, use U to find $K_L(x_t)$
3. If $K_L(x_t) \leq |C|$, keep going
4. Else return x_t, call this z, and stop

So $K_L(z) > |C|$, but C produces z and stops: contradiction
Due to [5], cited by [6].

There is no algorithm to *approximate* $K_L(x)$
In particular, `gzip` does not approximate $K_L(x)$
Can never say: x is incompressible
Can say: most things are random
don't know x *isn't* random yet

Mean Algorithmic Information and Entropy Rate

For an IID source

$$\lim_{n\to\infty} \frac{1}{n}\mathbf{E}\left[K(X_1^n)\right] = H[X_1]$$

For a general stationary source

$$\lim_{n\to\infty} \frac{1}{n}\mathbf{E}\left[K(X_1^n)\right] = h_1$$

Algorithmic Information of Dynamical Systems

Kolmogorov complexity of a continuous state:

$$K(s) = \sup_{\mathcal{B}} \lim_{n \to \infty} \frac{1}{n} K(b_1^n(s))$$

with

$$b_1^n(s) = b(s), b(\Phi(s)), b(\Phi^2(s)), \ldots b(\Phi^n(s))$$

If Φ is ergodic, then
BRUDNO'S THEOREM: with probability 1,

$$K(s) = h_{KS}$$

independent of the initial state s

In every *testable* way, typical chaotic symbol sequences look like they're generated by a stochastic process

Once again:

ANY DETERMINISM DISTINGUISHABLE FROM RANDOMNESS IS INSUFFICIENTLY COMPLEX

The key is the sensitive dependence on initial conditions:

$$h_{KS} \leq \sum_{i=1}^{d} \lambda_i \mathbf{1}_{x>0}(\lambda_i)$$

Instability reads out the algorithmic information which went into the initial condition

Why Does Attractor Reconstruction Need Determinism?

Attractor reconstruction only works if the attractor has finite dimension

A random process is basically an infinite-dimensional dynamical system

Use the shift-map representation

Attractor reconstruction breaks down when used on stochastic processes

Hand-waving about continuous variables

There's a theory of universal computation on real numbers & such

See Prof. Lenore Blum in SCS [7, 8, 9]

Or see Cris Moore

Works basically like discrete theory

Incompressibility results still there (more or less)

So $\exists$ incompressible sequences of continuous values

These come from chaotic infinite-dimensional dynamics

But: don't know of rigorous proofs

What About Finite-Dimensional Dynamics?

Three kinds of results:

1. About ensemble distributions, as in mixing
2. About projections — if we ignore some coordinates, the others look like a stochastic process
3. About approximations — real trajectories are close to those of stochastic processes [10]

Always *some* departures from randomness *if* we can see exact state

E.g., always *some* function of s_t which gives us $s_{t+10^{100}}$

even $s_{t+10^{10^{100}}}$

but function becomes harder and harder to evaluate, needs more and more data

This gets into subtle topics in approximation theory

Summing Up

Probability tells us what random processes look like
Incompressibility gives us realizations of those theories
Coarse-graining of unstable deterministic dynamics gives us incompressibility
Randomness can be produced by fully deterministic processes
Stochastic modeling works even in a fully deterministic *but chaotic* world

ANY SIGNAL DISTINGUISHABLE FROM NOISE IS INSUFFICIENTLY COMPRESSED

ANY DETERMINISM DISTINGUISHABLE FROM RANDOMNESS IS INSUFFICIENTLY COMPLEX

TOO MUCH OF A GOOD THING IS WONDERFUL

[1] David Ruelle. *Chance and Chaos*. Princeton University Press, Princeton, New Jersey, 1991.

[2] Ming Li and Paul M. B. Vitányi. *An Introduction to Kolmogorov Complexity and Its Applications*. Springer-Verlag, New York, second edition, 1997.

[3] Remo Badii and Antonio Politi. *Complexity: Hierarchical Structures and Scaling in Physics*. Cambridge University Press, Cambridge, England, 1997.

[4] Henri Poincaré. *The Value of Science: Essential Writings of Henri Poincaré*. Modern Library, New York, 2001. Contents: *Science and Hypothesis* (1903, trans. 1905); *The Value of Science* (1905, trans. 1913); *Science and Method* (1908; trans. 1914).

[5] R. Nohre. *Some topics in descriptive complexity*. PhD thesis, Linkoping University, Linkoping, Sweden, 1994.

[6] Jorma Rissanen. Complexity and information in data. In Andreas Greven, Gerhard Keller, and Gerald Warnecke,

editors, *Entropy*, Princeton Series in Applied Mathematics, pages 299–312, Princeton, New Jersey, 2003. Princeton University Press.

[7] Lenore Blum. Computing over the reals: Where Turing meets Newton. *Notices of the American Mathematical Society*, 51:1024–1034, 2004. URL `http://www.cs.cmu.edu/~lblum/PAPERS/TuringMeetsNewton.pdf`.

[8] Lenore Blum. Lectures on a theory of computation and complexity over the reals (or an arbitrary ring). In Erica Jen, editor, *1989 Lectures in Complex Systems*, volume 2 of *Santa Fe Institute Studies in the Sciences of Complexity (Lectures)*, pages 1–48, Redwood City, California, 1990. Addison-Wesley.

[9] Lenore Blum, Michael Shub, and Steven Smale. On a theory of computation and complexity over the real numbers: NP-completeness, recursive functions and

universal machines. *Bulletin of the American Mathematical Society*, 21:1–46, 1989.

[10] Gregory L. Eyink. Linear stochastic models of nonlinear dynamical systems. *Physical Review E*, 58:6975–6991, 1998.