Quantitative Complexity Measures

Cosma Shalizi

16 June 2010
Complex Systems Summer School
Complexity Measures

“Complex” \( \approx \) “many strongly interacting effective degrees of freedom”
So not: only a few variables; most independent variables; lots of variables but only a few are relevant
Can we quantify this idea?
If so, what is the number good for?
I often say that when you can measure what you are speaking about, and express it in numbers, you know something about it; but when you cannot express it in numbers, your knowledge is of a meagre and unsatisfactory kind; it may be the beginning of knowledge, but you have scarcely, in your thoughts, advanced to the stage of science, whatever the matter may be. — Thermodynamicist W. Thomson, a.k.a. Lord Kelvin
I often say that when you can measure what you are speaking about, and express it in numbers, you know something about it; but when you cannot express it in numbers, your knowledge is of a meagre and unsatisfactory kind; it may be the beginning of knowledge, but you have scarcely, in your thoughts, advanced to the stage of science, whatever the matter may be. — Thermodynamicist W. Thomson, a.k.a. Lord Kelvin

but quantifying the wrong things advances a meagre and unsatisfactory understanding to the stage of pseudoscience, like IQ testing
most complexity measures are “conspicuously vacuous” (Landauer, 1988)
most complexity measures are “conspicuously vacuous” (Landauer, 1988)

*The urge to destroy is also a creative urge.*

— *Distributed systems theorist M. Bakunin*
Three Kinds of Complexity

1. **Description** of the system, in the preferred or optimal model (units: bits)
   Wiener, von Neumann, Kolmogorov, Pagels and Lloyd, . . .

2. **Learning** that model (samples)
   Fisher, Neyman, Reichenbach, Vapnik and Chervonenkis, Valiant, . . .

3. **Computational** complexity of the model (units: ops)
   These are (pretty much) orthogonal
   I will focus on description, with an occasional glance at learning
Badii and Politi (1997)
Badii and Politi (1997)
Feldman and Crutchfield (1998)
General references

Badii and Politi (1997)
Feldman and Crutchfield (1998)
Shalizi and Crutchfield (2001, appendices), Shalizi (2006, §8)
(discount appropriately)
What We Would Like

Low values for easily described determinism
What We Would Like

Low values for easily described determinism
Low values for easily described IID randomness
What We Would Like

Low values for easily described determinism
Low values for easily described IID randomness
High values for lots of strong interactions, lots of heterogeneity, lots of consequential options
What We Would Like

Low values for easily described determinism
Low values for easily described IID randomness
High values for lots of strong interactions, lots of heterogeneity, lots of consequential options
Number should have implications about other stuff
Compression

Ordinary information theory: concise description of random objects
Can also think about coding and compression of particular objects, without reference to a generating distribution
**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
Compression by Programming

Lossless compression needs an effective procedure — definite steps which a machine could take to recover the original
Effective procedures = algorithms
Algorithms = recursive functions
Recursive functions = Turing machines
finite automaton with an unlimited external memory
Think about programs written in a universal language (R, Lisp, Fortran, C, C++, Pascal, Java, Perl, OCaml, Forth, ...)

$x$ is our object, size $|x|$
$x$ is our object, size $|x|$
Desired: a program in language $L$ which will output $x$ and then stop
those programs are descriptions of $x$
$x$ is our object, size $|x|$

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

those programs are descriptions of $x$

What is the *shortest* program which will do this?
$x$ is our object, size $|x|$
Desired: a program in language $L$ which will output $x$ and then stop
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
\(x\) is our object, size \(|x|\)

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

those programs are descriptions of \(x\)

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

N.B.: \texttt{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 \( L \) being universal?

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

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

where $\mathcal{D}(x) = \text{all programs in } L \text{ that output } x \text{ and then halt}$

This is the **algorithmic information content** of $x$
Kolmogorov Complexity

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

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

where $\mathcal{D}(x) = \text{all programs in } L \text{ that output } x \text{ and then halt}$

This is the **algorithmic information content** of $x$

a.k.a. Kolmogorov-Chaitin complexity,
Kolmogorov-Chaitin-Solomonoff complexity...
The **Kolmogorov complexity** of $x$, relative to $L$, is

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

where $D(x) =$ 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.
The **Kolmogorov complexity** of $x$, relative to $L$, is

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

where $\mathcal{D}(x) = \text{all programs in } L \text{ that output } x \text{ and then halt}$

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”: \( K \leq 1 + c \)
Examples

“0”: \( K \leq 1 + c \)

“0” ten thousand times: \( K \leq 1 + \log_2 10^4 + c = 1 + 4 \log_2 10 + c \)
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: $K \leq 1 + 10 \log_2 10 + c$
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: $K \leq 8 + 10 \log_2 10 + c$
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: \( K \leq 8 + 10 \log_2 10 + c \)

\( \pi \), first \( n \) digits: \( K \leq g + \log_2 n \)
“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: $K \leq 8 + 10 \log_2 10 + c$
$\pi$, first $n$ digits: $K \leq g + \log_2 n$

In fact, any number you care to name contains little algorithmic information

Why?
Most Random Sequences are Incompressible

Most objects are not very compressible
Most Random Sequences are Incompressible

Most objects are not very compressible
Exactly $2^n$ objects of length $n$ bits
Most objects are not very compressible
Exactly $2^n$ objects of length $n$ bits
At most $2^k$ programs of length $k$ bits
Most Random Sequences are Incompressible

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
Most Random Sequences are Incompressible

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 $\leq 2^{k-n}$
Most Random Sequences are Incompressible

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
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
Most Random Sequences are Incompressible

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
Mean Algorithmic Information and Entropy Rate

For an IID source

$$\lim_{n \to \infty} \frac{1}{n} \mathbb{E}[K(X_1^n)] = H[X_1]$$
Mean Algorithmic Information and Entropy Rate

For an IID source

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

For a general stationary source

$$\lim_{n \to \infty} \frac{1}{n} E[K(X_1^n)] = h_1$$
Mean Algorithmic Information and Entropy Rate

For an IID source

\[
\lim_{n \to \infty} \frac{1}{n} \mathbb{E}[K(X_1^n)] = H[X_1]
\]

For a general stationary source

\[
\lim_{n \to \infty} \frac{1}{n} \mathbb{E}[K(X_1^n)] = h_1
\]

also (with more conditions) \( n^{-1} K(X_1^n) \to h_1 \) in probability
Why You Should Not Use Algorithmic Information As Your Complexity Measure
Why You Should Not Use Algorithmic Information As Your Complexity Measure

1. You can’t figure out what it is
Why You Should Not Use Algorithmic Information As Your Complexity Measure

1. You can’t figure out what it is
2. Even if you could, it doesn’t do what you want
Kolmogorov Complexity Is Uncomputable

There is no algorithm to compute $K_L(x)$
Kolmogorov Complexity Is Uncomputable

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 $V$ which compresses the incompressible:

1. Sort all sequences by length and then alphabetically
2. For the $i^{\text{th}}$ sequence $x^{(i)}$, use $U$ to find $K_L(x^{(i)})$
3. If $K_L(x^{(i)}) \leq |V|$, keep going
4. Else set $z$ to $x^{(i)}$, return $z$, and stop

Kolmogorov Complexity Is Uncomputable

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 $V$ which compresses the incompressible:

1. Sort all sequences by length and then alphabetically
2. For the $i^{th}$ sequence $x^{(i)}$, use $U$ to find $K_L(x^{(i)})$
3. If $K_L(x^{(i)}) \leq |V|$, keep going
4. Else set $z$ to $x^{(i)}$, return $z$, and stop

So $K_L(z) > |V|$, but $V$ outputs $z$ and stops: contradiction

Kolmogorov Complexity Is Uncomputable

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 $V$ which compresses the incompressible:

1. Sort all sequences by length and then alphabetically
2. For the $i^{th}$ sequence $x^{(i)}$, use $U$ to find $K_L(x^{(i)})$
3. If $K_L(x^{(i)}) \leq |V|$, keep going
4. Else set $z$ to $x^{(i)}$, return $z$, and stop

So $K_L(z) > |V|$, but $V$ outputs $z$ and stops: contradiction
There is no algorithm to \textit{approximate} $K_L(x)$
There is no algorithm to *approximate* $K_L(x)$
In particular, $gzip$ does not approximate $K_L(x)$
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: haven’t managed to compress $x$ yet
Incompressible Sequences Look Random

Suppose $x$ is a binary string of length $n$, with $n \gg 1$
Incompressible Sequences Look Random

Suppose \( x \) is a binary string of length \( n \), with \( n \gg 1 \)

If proportion of 1s in \( x \) is \( p \), then (Exercise)

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

*Hint:* Use Stirling’s formula to count the number of strings
Incompressible Sequences Look Random

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

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

*Hint:* Use Stirling’s formula to count the number of strings

$nH(p) < n$ if $p \neq \frac{1}{2}$
Incompressible Sequences Look Random

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

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

*Hint: Use Stirling’s formula to count the number of strings

$nH(p) < n$ if $p \neq \frac{1}{2}$
Similarly for statistics of pairs, triples, ...
Suppose $x$ is a binary string of length $n$, with $n \gg 1$. If the proportion of 1s in $x$ is $p$, then (exercise)

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

*Hint:* Use Stirling’s formula to count the number of strings

$nH(p) < n$ 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
Suppose $x$ is a binary string of length $n$, with $n \gg 1$

If proportion of 1s in $x$ is $p$, then (EXERCISE)

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

*Hint:* Use Stirling’s formula to count the number of strings

$nH(p) < n$ 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**
CLAIM 1: Incompressible sequences have all the *effectively testable* properties of pure noise.
Incompletely Sequences Look Random (Cont.)

Claim 1: Incompletely 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
Incomprssible Sequences Look Random (Cont.)

**Claim 1**: Incomprssible 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
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,

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

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

Power of this test is close to that of any other (computable) test (Martin-Lof)
Why the $L$ doesn’t matter

Take your favorite sequence $x$
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 $L$. 
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 $L$
Makes $K_{L'}(x) = 1$, but makes descriptions of other strings 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 (2001) said as much 100 years ago, without the math

Excerpt on website

Extends to other, partially-compressible stochastic processes
The maximally-compressed description is incompressible
so other stochastic processes are transformations of noise
“Occam’s Razor” theorem: If your model can be written as a short program and it does well on training data, then it will probably generalize well to new data.
“Occam’s Razor” theorem: If your model can be written as a short program and it does well on training data, then it will probably generalize well to new data. This is a total cheat; works because there just aren’t many short programs; any other sparse set of models will do.
“Occam’s Razor” theorem: If your model can be written as a short program and it does well on training data, then it will probably generalize well to new data. This is a total cheat; works because there just aren’t many short programs; any other sparse set of models will do say ones whose lengths are exactly $k^{k^k}$, $k$ prime and $< |x|$
“Occam’s Razor” theorem: If your model can be written as a short program and it does well on training data, then it will probably generalize well to new data. This is a total cheat; works because there just aren’t many short programs; any other sparse set of models will do say ones whose lengths are exactly $k^k$, $k$ prime and $< |x|$.

For much better ideas on Occam’s Razor, see http://www.andrew.cmu.edu/user/kk3n/ockham/Ockham.html
Sophistication

Gács et al. (2001)
Gács et al. (2001)

Separate the minimal program into an algorithm and input data
Gács et al. (2001)

Separate the minimal program into an algorithm and input data
\[ \text{Soph}(x) \equiv \text{length of shortest algorithm for which } x \text{ is a “typical” output} \]

Tricky definition of “typical”
Gács et al. (2001)

Separate the minimal program into an algorithm and input data
\[ \text{Soph}(x) \equiv \text{length of shortest algorithm for which } x \text{ is a “typical” output} \]

Tricky definition of “typical”

*Not* just randomness
Gács et al. (2001)

Separate the minimal program into an algorithm and input data
$\text{Soph}(x) \equiv \text{length of shortest algorithm for which } x \text{ is a “typical” output}$

Tricky definition of “typical”

*Not* just randomness

Interesting predictive consequences (“algorithmic sufficient statistics”)
Gács *et al.* (2001)

Separate the minimal program into an algorithm and input data

\[ \text{Soph}(x) \equiv \text{length of shortest algorithm for which } x \text{ is a “typical” output} \]

Tricky definition of “typical”

*Not* just randomness

Interesting predictive consequences (“algorithmic sufficient statistics”)

Still completely uncomputable
Logical Depth


Logical depth of $x \approx$ how long does the shortest program for $x$ take to run?
If $K_L(x)$ is small but many operations are required, deeper than
if $K_L(x) \approx |x|$ but so is the run-time
∴ random strings could be shallower than say $\pi$
Logical Depth


Logical depth of $x \approx$ how long does the shortest program for $x$ take to run?
If $K_L(x)$ is small but many operations are required, deeper than if $K_L(x) \approx |x|$ but so is the run-time
∴ random strings could be shallower than say $\pi$
Still completely uncomputable
Morals from Kolmogorov Complexity

We don’t *just* want to measure randomness; we’ve got entropy for that
A good complexity measure should be low for noise
Morals from Kolmogorov Complexity

We don’t *just* want to measure randomness; we’ve got entropy for that
A good complexity measure should be low for noise
“To describe coin tosses, toss a coin”
We don’t *just* want to measure randomness; we’ve got entropy for that
A good complexity measure should be low for noise
“To describe coin tosses, toss a coin”
A good complexity measure should be something we can actually calculate
Morals from Kolmogorov Complexity

We don’t *just* want to measure randomness; we’ve got entropy for that.

A good complexity measure should be low for noise.

“To describe coin tosses, toss a coin.”

A good complexity measure should be something we can actually calculate.

Best reference on Kolmogorov complexity: Li and Vitányi (1997)
Thermodynamic Depth

Lloyd and Pagels (1988)

Thermodynamic depth = Shannon entropy of trajectories leading to the current state
How many bits do we need to describe the particular history that assembled this state (given that it did end up here)?
Simple states have easily-described histories
Complex states have histories that need lots of information
Alas: depth grows to infinity in a stationary process
See Crutchfield and Shalizi (1999)
Recall from last time:

- A statistic (function of the history) $\epsilon$ is **sufficient** when $I[X_{t+1}^\infty; X_{-\infty}^t] = I[X_{t+1}^\infty; \epsilon(X_{-\infty}^t)]$

- A sufficient statistic is **minimal** when $\epsilon = g(\eta)$ for any other sufficient $\eta$, thus $I[X_{-\infty}^t; \epsilon(X_{-\infty}^t)] \leq I[X_{-\infty}^t; \eta(X_{-\infty}^t)]$

- Minimal sufficient statistics are unique (up to re-labeling of values)

- We can construct them and (sometimes) estimate them
Statistical Complexity

Definition

\[ C_{GCY} \equiv I[\epsilon(X^t_{-\infty}); X^t_{-\infty}] \text{ is the statistical forecasting complexity of the process} \]
Statistical Complexity

Definition

\[ C_{GCY} \equiv I(\epsilon(X^t_{-\infty}); X^t_{-\infty}) \] is the statistical forecasting complexity of the process

= amount of information about the past needed for optimal prediction
Statistical Complexity

**Definition**

\[ C_{GCY} \equiv I[\epsilon(X_{-\infty}^t); X_{-\infty}^t] \]

is the **statistical forecasting complexity** of the process

= amount of information about the past needed for optimal prediction

Verbal formulation from Grassberger (1986)
Statistical Complexity

Definition

\[ C_{GCY} \equiv I[\epsilon(X^t_{-\infty}); X^t_{-\infty}] \] is the statistical forecasting complexity of the process

- amount of information about the past needed for optimal prediction

Verbal formulation from Grassberger (1986)
Crutchfield and Young (1989) made “state” and “optimal prediction” precise
Definition

$$C_{GCY} \equiv I[\epsilon(X_t^\infty); X_t^\infty]$$ is the statistical forecasting complexity of the process

= amount of information about the past needed for optimal prediction

Verbal formulation from Grassberger (1986)
Crutchfield and Young (1989) made “state” and “optimal prediction” precise
Split the difference and call it GCY complexity
Some Properties of GCY Complexity

Grows with the diversity of statistically distinct patterns of behavior

\[ = H[\epsilon(X_{-\infty}^t)] \text{ for discrete causal states} \]
Some Properties of GCY Complexity

Grows with the diversity of statistically distinct patterns of behavior

\[ H[\epsilon(X_{-\infty}^t)] \] for discrete causal states

\[ = \text{average-case sophistication} \]
Some Properties of GCY Complexity

Grows with the diversity of statistically distinct patterns of behavior

\[ H[\epsilon(X^t_{-\infty})] \] for discrete causal states

\[ = \text{average-case sophistication} \]

\[ = \log(\text{period}) \] for period processes

\[ = \log(\text{geometric mean(recurrence time)}) \] for stationary processes

\[ = \text{information about microstate in macroscopic observations} \] (sometimes)
Some Properties of GCY Complexity

Grows with the diversity of statistically distinct patterns of behavior

\[ = H[\epsilon(X_{-\infty}^t)] \] for discrete causal states

\[ = \text{average-case sophistication} \]

\[ = \log(\text{period}) \] for period processes

\[ = \log(\text{geometric mean(remurrence time)}) \] for stationary processes
Some Properties of GCY Complexity

Grows with the diversity of statistically distinct patterns of behavior

\[ H[\epsilon(X_{-\infty}^t)] \] for discrete causal states

\[ = \text{average-case sophistication} \]

\[ = \log(\text{period}) \] for period processes

\[ = \log(\text{geometric mean(recurrence time)}) \] for stationary processes

\[ = \text{information about microstate in macroscopic observations} \]

(sometimes)
Predictive Information

\[ I_{\text{pred}} \equiv I[X_{t+1}^\infty; X_t^{\infty}] \]
Predictive Information

\[ I_{\text{pred}} \equiv I[X_{t+1}^\infty; X_t^{-\infty}] \]

a.k.a. effective measure complexity, excess entropy, …

Easily shown that

\[ I[X_{t+1}^\infty; X_t^t] = I[X_{t+1}^\infty; \epsilon(X_t^{-\infty})] \leq I[\epsilon(X_t^{-\infty}); X_t^t] \]
Predictive Information

\[ I_{\text{pred}} \equiv I[X_{t+1}^\infty; X_t^{-\infty}] \]

a.k.a. effective measure complexity, excess entropy, …

Easily shown that

\[ I[X_{t+1}^\infty; X_t^{-\infty}] = I[X_{t+1}^\infty; \epsilon(X_t^{-\infty})] \leq I[\epsilon(X_t^{-\infty}); X_t^{-\infty}] \]

You need at least \( m \) bits of state to get \( m \) bits of prediction
Predictive Information

\[ I_{\text{pred}} \equiv I[X_{t+1}^\infty; X_t^{-\infty}] \]

a.k.a. effective measure complexity, excess entropy, …

Easily shown that

\[ I[X_{t+1}^\infty; X_t^{-\infty}] = I[X_{t+1}^\infty; \epsilon(X_t^{-\infty})] \leq I[\epsilon(X_t^{-\infty}); X_t^{-\infty}] \]

You need at least \( m \) bits of state to get \( m \) bits of prediction

Efficiency of prediction = \( I_{\text{mathrmpred}} / C_{GCY} \leq 1 \)
Spatio-Temporal Prediction

Dynamic random field: $X(\vec{r}, t)$
Assume a finite “speed of light”
Past light cone of $(\vec{r}, t)$: all points at earlier times from which a signal could have come
Future light cone: all points at later times to which a signal could go
Light cones in $1 + 1D$
Local Causal States

Go through equivalence classing again, only now for predicting the configuration in the future cone from that in the past cone. Still minimal sufficient statistics, recursive updating (on new information), local states form a Markov random field (Shalizi, 2003; Shalizi et al., 2004, 2006)
The system self-organizes between time $t_1$ and $t_2$ iff (1) $C(t_2) > C(t_1)$, and (2) this increase is not all externally caused.
Self-Organization

The system self-organizes between time $t_1$ and $t_2$ iff (1) $C(t_2) > C(t_1)$, and (2) this increase is not all externally caused. (2) is the problem of exorcising demons.
Emergence

Start with a process \((X_t)\) at one level of description, get \(C(X)\), \(I_{\text{pred}}(X)\)
Start with a process \((X_t)\) at one level of description, get \(C(X)\), \(I_{\text{pred}}(X)\).

Coarse-grain it to get a higher level (more abstract, less refined) description, with induced process \((Y_t)\), with its own \(C(Y)\), \(I_{\text{pred}}(Y)\).
Emergence

Start with a process \((X_t)\) at one level of description, get \(C(X)\), \(I_{\text{pred}}(X)\).

Coarse-grain it to get a higher level (more abstract, less refined) description, with induced process \((Y_t)\), with its own \(C(Y)\), \(I_{\text{pred}}(Y)\).

Higher level emerges iff

\[
\frac{I_{\text{pred}}(Y)}{C(Y)} > \frac{I_{\text{pred}}(X)}{C(X)}
\]
Emergence

Start with a process \((X_t)\) at one level of description, get \(C(X)\), \(I_{\text{pred}}(X)\)

Coarse-grain it to get a higher level (more abstract, less refined) description, with induced process \((Y_t)\), with its own \(C(Y)\), \(I_{\text{pred}}(Y)\)

Higher level emerges iff

\[
\frac{I_{\text{pred}}(Y)}{C(Y)} > \frac{I_{\text{pred}}(X)}{C(X)}
\]

Can e.g. show that thermodynamic descriptions emerge from statistical-mechanical ones (Shalizi and Moore, 2003)
Local Statistical Complexity

Shalizi et al. (2006)

\[ C(\vec{r}, t) \equiv - \log \Pr (S = s(\vec{r}, t)) \]
Local Statistical Complexity

Shalizi et al. (2006)

\[ C(\vec{r}, t) \equiv - \log \Pr (S = s(\vec{r}, t)) \]

Gives the local density of the information needed for prediction
Local Statistical Complexity

Shalizi et al. (2006)

\[ C(\vec{r}, t) \equiv -\log \Pr (S = s(\vec{r}, t)) \]

Gives the local density of the information needed for prediction
Can change over space and time
Local Statistical Complexity

Shalizi et al. (2006)

\[
C(\vec{r}, t) \equiv -\log Pr \left( S = s(\vec{r}, t) \right)
\]

Gives the local density of the information needed for prediction
Can change over space and time
Use to automatically filter for the interesting bits
Cyclic Cellular Automata, as an Example

Quantitative model of excitable media
\( \kappa \) colors; a cell of color \( k \) switches to \( k + 1 \mod \kappa \) if at least \( T \) neighbors are already of that color
Analytical theory for structures formed (Fisch et al., 1991a,b)
Generic behaviors: spirals, “turbulence”, local oscillations, fixation
Initial configuration, $T = 1$
Final configuration, $T = 1$ (oscillates forever)
Initial configuration, $T = 4$
Final configuration, $T = 4$ (static blocks)
Initial configuration, $T = 2$
Intermediate time configuration, $T = 2$
Asymptotic configuration, $T = 2$, rotating spirals
Initial configuration, $T = 3$
Intermediate time configuration, $T = 3$
Asymptotic configuration, $T = 3$, turbulent seething gurp
$C_{GCY}$ vs. time and threshold, 300 $\times$ 300 lattice, 30 replicas
Typical long-time configuration
Hand-crafted order parameter field
Local complexity field
Order parameter (broken symmetry, physical insight, tradition, trial and error, current configuration) vs. local statistical complexity (prediction, automatic, time evolution)
Streamlines from computational fluid dynamics; color indicates local complexity of velocity field (Jänicke et al., 2007)
Zombie Complexities

Ideas which should be dead, but continue to eat brains

- Prigogine’s ideas on dissipative structures
- Haken’s synergetics
- Wolfram’s 4 classes of CA
- The edge of chaos — see Mitchell *et al.* (1993)
- \((\text{disorder}) \times (1 - \text{disorder})\) — see Binder and Perry (2000); Crutchfield *et al.* (2000)
- Self-organized criticality (as a ruling idea)
- Power-laws, *therefore* complex
- Tsallis statistics
Why Physicists Think Power Laws Are Cool

Go back to fundamental statistical mechanics
Macroscopic variable $M =$ coarse-graining of microscopic state
$W(m) =$ volume of microstates $x$ such that $M(x) = m$
Boltzmann entropy $S_B(m) = \log W(m)$
Equilibrium $=$ state $m^*$ maximizing $S_B$
Einstein formula for fluctuations around equilibrium:

$$\Pr (M = m) \propto e^{S_B(m)}$$
Expand around $m^*$, so $\partial S_B / \partial m = 0$ at $m^*$

$$\Pr(M = m) \propto e^{S(m^*) + \frac{1}{2} \frac{\partial^2 S(m^*)}{\partial m^2} (m - m^*)^2 + \text{h.o.t.}}$$

$$\propto e^{\frac{1}{2} \frac{\partial^2 S(m^*)}{\partial m^2} (m - m^*)^2 + \text{h.o.t.}}$$

drop the h.o.t.

$$M \sim \mathcal{N}(m^*, -\frac{\partial^2 S(m^*)}{\partial m^2})$$
correlations are short range
⇒ rapid approach to independence, exponential mixing
⇒ central limit theorem for averages over space (and time)
⇒ Gaussians for macroscopic variables (which are averages)
Phase Transitions

See Yeomans (1992) for nice introduction
Basically, bifurcations: behavior changes suddenly as temperature (or pressure or other control variable) crosses some threshold
First order: entropy is discontinuous at critical point
Examples: ice/water at 273K (and standard pressure); water/steam at 373K
order parameter is discontinuous
Second order: derivative of entropy is discontinuous
Example: “Curie point”, permanent magnetization/not in iron 1043K
order parameter continuous but with sharp kink
like amplitude of limit cycle in period-doubling
Focus on continuous (second-order) case
Critical fluctuations

Entropy story breaks down because derivatives $\to \pm \infty$

Competition between two phases, no preference, one can pop up in the middle of the other

Fluctuations get arbitrarily large $\Rightarrow$ long-range correlations $\Rightarrow$

slow mixing (if any)

Assemblage becomes self-similar: magnify a small part and it looks just like the whole thing ("renormalization")

only strictly true for infinitely big assemblages

averaging must lead to a self-similar distribution

Power laws are self-similar (scale-free)

Conclusion: at critical point, expect to see power law distributions
Theory of phase transitions / critical phenomena / order parameters / renormalization one of the key developments in physics over the last half century (Yeomans, 1992; Domb, 1996)
Theory of phase transitions / critical phenomena / order parameters / renormalization one of the key developments in physics over the last half century (Yeomans, 1992; Domb, 1996)

⇒ physicists think criticality is Very Cool
Theory of phase transitions / critical phenomena / order parameters / renormalization one of the key developments in physics over the last half century (Yeomans, 1992; Domb, 1996)
⇒ physicists think criticality is Very Cool
Criticality ⇒ power law distributions
Theory of phase transitions / critical phenomena / order parameters / renormalization one of the key developments in physics over the last half century (Yeomans, 1992; Domb, 1996)
⇒ physicists think criticality is Very Cool
Criticality ⇒ power law distributions
so physicists tend to think:
Theory of phase transitions / critical phenomena / order parameters / renormalization one of the key developments in physics over the last half century (Yeomans, 1992; Domb, 1996)
⇒ physicists think criticality is Very Cool
Criticality ⇒ power law distributions
so physicists tend to think:

(i) ¬ power laws ⇒ ¬ critical ⇒ Bored Now
Theory of phase transitions / critical phenomena / order parameters / renormalization one of the key developments in physics over the last half century (Yeomans, 1992; Domb, 1996)
⇒ physicists think criticality is Very Cool
Criticality ⇒ power law distributions
so physicists tend to think:

(i) ¬ power laws ⇒ ¬ critical ⇒ Bored Now
(ii) power laws ⇒ critical → Very Cool
Theory of phase transitions / critical phenomena / order parameters / renormalization one of the key developments in physics over the last half century (Yeomans, 1992; Domb, 1996)
⇒ physicists think criticality is Very Cool
Criticality ⇒ power law distributions
so physicists tend to think:

(i) ¬ power laws ⇒ ¬ critical ⇒ Bored Now
(ii) power laws ⇒ critical → Very Cool

(ii) is called “the fallacy of affirming the consequent”
Many ways to get power laws or other heavy-tailed distributions
Many ways to get power laws or other heavy-tailed distributions e.g., exponential growth for a random time (Reed and Hughes, 2002)
Many ways to get power laws or other heavy-tailed distributions e.g., exponential growth for a random time (Reed and Hughes, 2002) or multiplicative fluctuations (Simon, 1955)
Tsallis Statistics

Take the MaxEnt procedure, but instead maximize

\[
H_q[X] \equiv \frac{1}{q - 1} \left( 1 - \sum_x (\Pr(X = x))^q \right)
\]

(similar form for continuous case)
Reverts to Shannon entropy as \( q \to 1 \)
leads to “\( q \)-exponential” CDF

\[
P_{q,\kappa}(X \geq x) = \left( 1 - \frac{(1 - q)x}{\kappa} \right)^{1/(1-q)}
\]
**q-Exponentials**

(Shalizi, 2007) Set

\[ q = 1 + \frac{1}{\theta}, \quad \kappa = \frac{\sigma}{\theta} \]

Observe

\[ P_{\theta,\sigma}(X \geq x) = (1 + x/\sigma)^{-\theta} \]

vs. “type II generalized Pareto distribution” (Arnold, 1983)

\[ P(X \geq x) = [1 + (x - \mu)/\sigma]^{-\alpha} \]

set \( \mu = 0 \) and \( \alpha = \theta \)

Comes from a mixture of exponentials (Maguire et al., 1952)
Tsallis statistics supposedly good for long-range interactions
Tsallis statistics supposedly good for long-range interactions but the MaxTsallisEnt principle doesn’t even agree with large deviations theory (La Cour and Schieve, 2000)
Tsallis statistics supposedly good for long-range interactions but the MaxTsallisEnt principle doesn’t even agree with large deviations theory (La Cour and Schieve, 2000) and large deviations *does* agree with the actual behavior of long-range interacting assemblages (Barré *et al.*, 2005)
Tsallis statistics supposedly good for long-range interactions but the MaxTsallisEnt principle doesn’t even agree with large deviations theory (La Cour and Schieve, 2000) and large deviations does agree with the actual behavior of long-range interacting assemblages (Barré et al., 2005) but Tsallis gives us power laws, so *Physica A* will love it forever and ever.
Tsallis statistics supposedly good for long-range interactions but the MaxTsallisEnt principle doesn’t even agree with large deviations theory (La Cour and Schieve, 2000) and large deviations does agree with the actual behavior of long-range interacting assemblages (Barré et al., 2005) but Tsallis gives us power laws, so *Physica A* will love it forever and ever.

If you want more:
http://bactra.org/notebooks/tsallis.html


