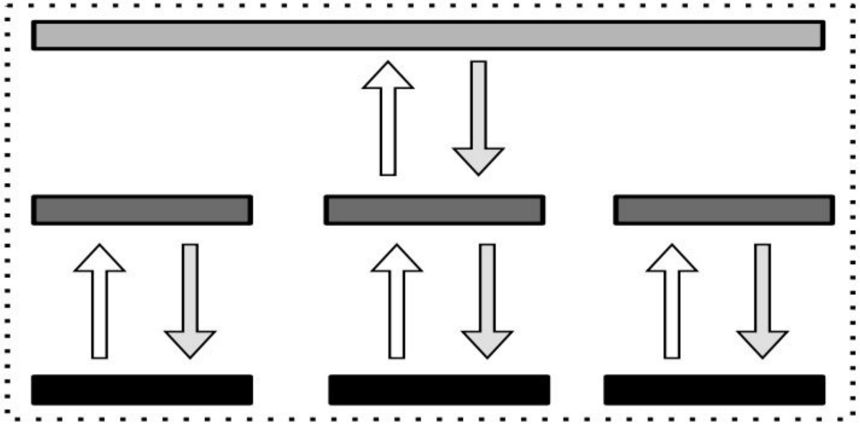
Complex Adaptive Systems

Interactions: Systems composed of interacting components

Emergence: Structure emerges from interactions among components and between components and their environment

Scale: Systems are nested and structure emerges at different scales

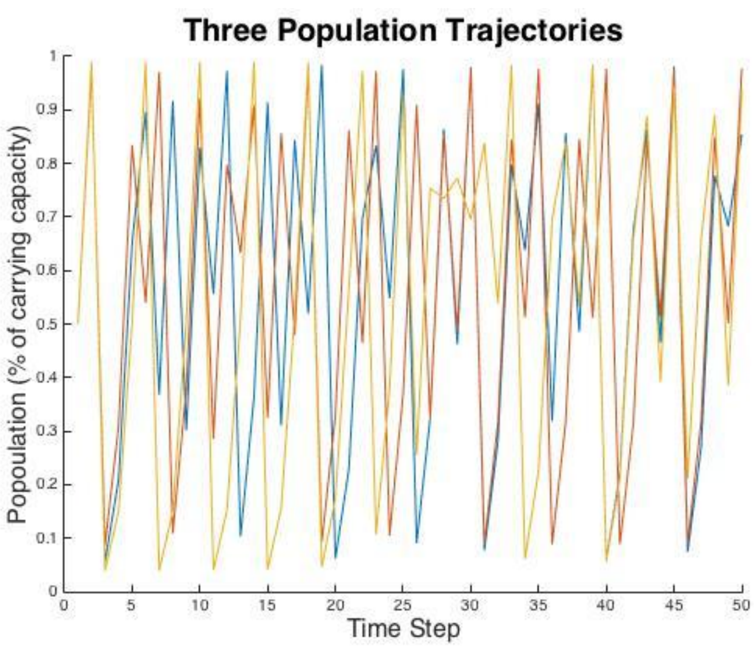
Evolution: Systems are dynamic and adapt to internal and external conditions

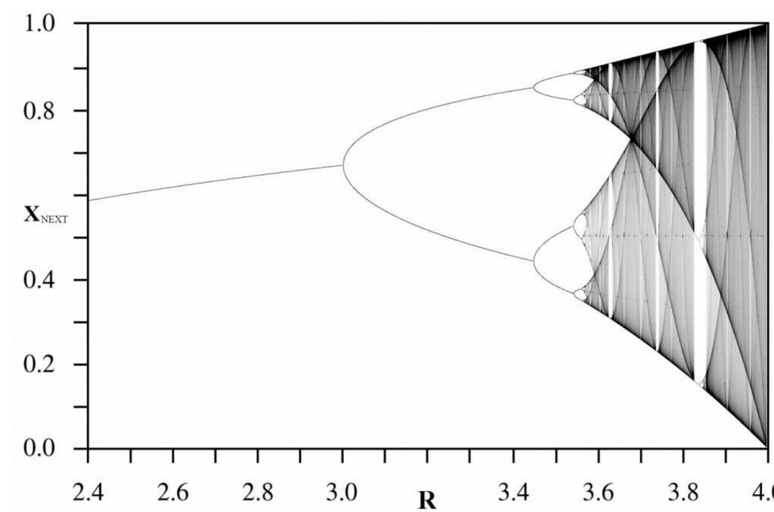


- Mitchell Ch 1: Complex Adaptive Systems
- Mitchell Ch 2: Logistic Map
  - Flake Chapter 10
- Mitchell Ch 3: Information Theory
- Mitchell Ch 4: Computation and self reference

Mitchell Ch 2, The Logistic Map: Chaos from a simple equation

$$x_{n+1} = rx_n(1 - x_n)$$





Logistic Map: A model of population growth

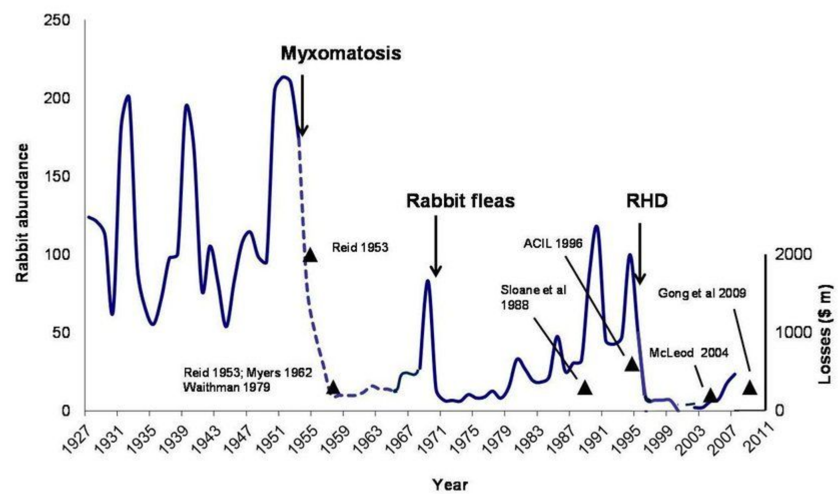
A European rabbit in Tasmania

Rabbits around a waterhole during myxomatosis trials, Wardang Island, South Australia, 1938. https://en.wikipedia.org/wiki/Myxomatosis

1859: 24 Rabbits successfully introduced into Australia

Australia is now home to at least 150 million feral rabbits

The proliferation of rabbits was the fastest of an introduced mammal anywhere in the world



https://www.nma.gov.au/defining-moments/resources/rabbits-introduced

Logistic Map

- Imagine a population of n rabbits on an island with carrying capacity of k rabbits.

- $n_{t+1} = (b\text{-}d)(n_t - n_t^2/k)$ As n approaches k, the growth slows
  - Let $x_t = n_t/k$ (relative population size-fraction of carrying capacity)

    $\mathbf{n_t = k\ x_t}$
  - Let **R = b-d** (population growth rate is birth – death rate)
  -

Logistic Map

- Imagine a population of n rabbits on an island with carrying capacity of k rabbits.

- $n_{t+1} = (b\text{-}d)(n_t - n_t^2/k)$ As n approaches k, the growth slows
  - Let $x_t = n_t/k$ (relative population size-fraction of carrying capacity)

    $\mathbf{n_t = k\, x_t}$
  - Let **R = b-d** (population growth rate is birth – death rate)

- $kx_{t+1} = R(kx_t - kx_t^2)$ ← this is just substituting in x_t and R

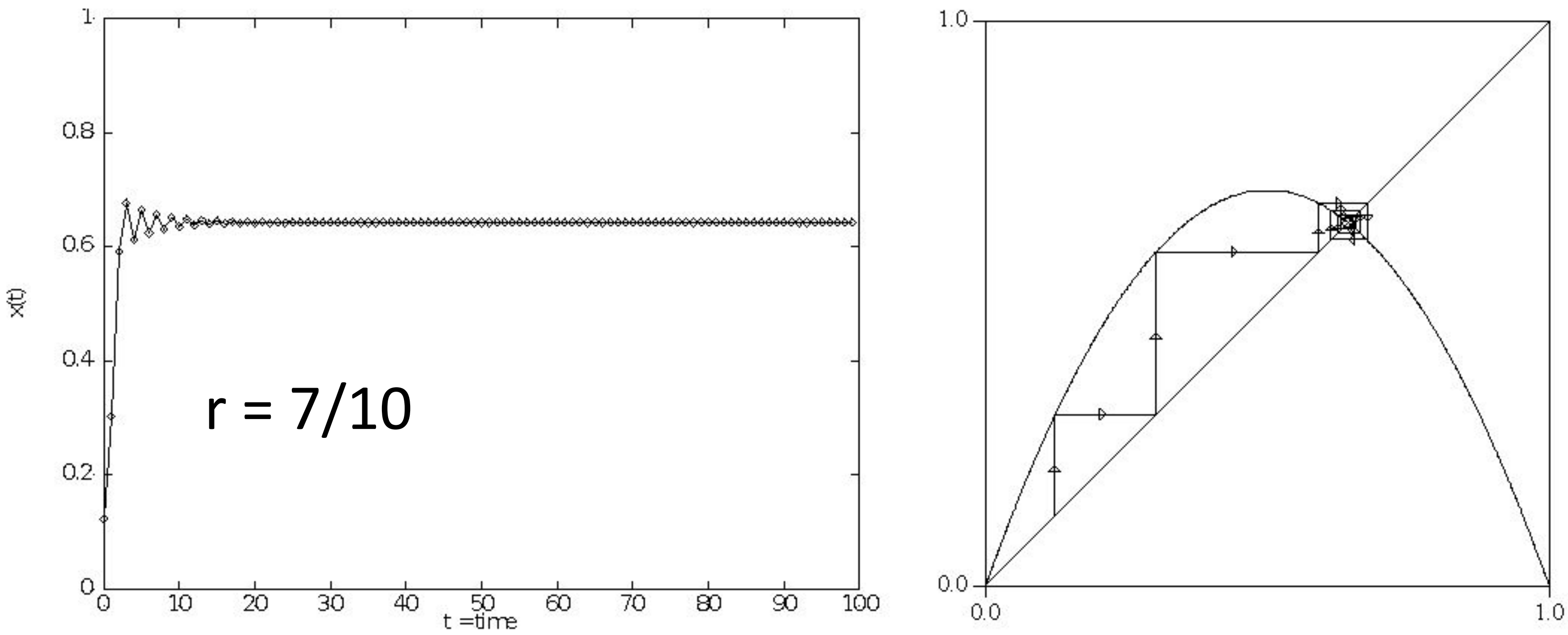
  $= Rk(x_t - x_t^2)$

$\mathbf{x_{t+1} = R\, x_t(1 - x_t)}$ **The logistic map**

(in Flake, CoBN $x_{t+1} = 4rx_t(1\text{-}x_t)$, so $R = 4r$)

$\mathbf{x_{t+1} = Rx_t(1 - x_t)}$ where R = 4r

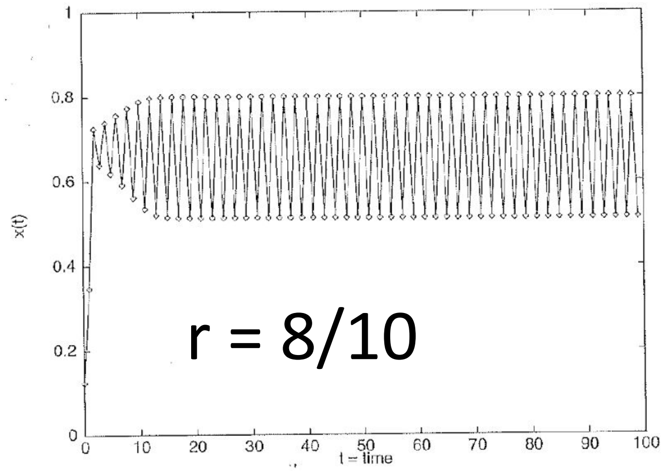
Return map:
$\mathbf{x_{t+1}}$ **vs.** $\mathbf{x_t}$



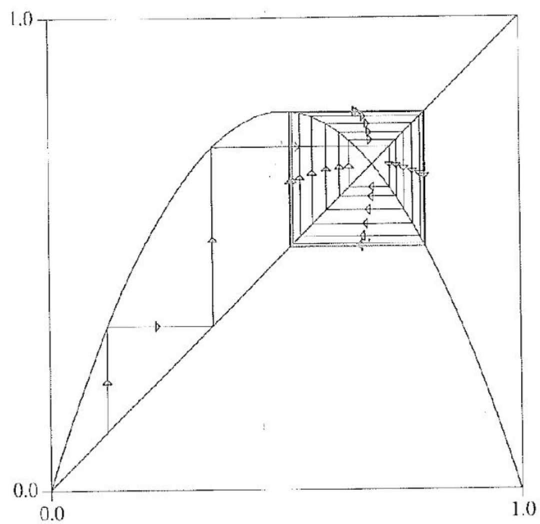
(a) (b)

Figure 10.2 Logistic map with $r = \frac{7}{10}$: (a) The time series quickly stabilizes to a fixed point. (b) The state space of the same system shows how subsequent steps of the system get pulled into the fixed point.

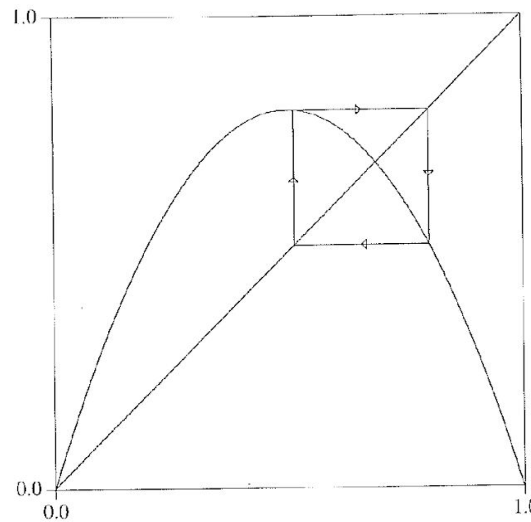
Figure from *The Computational Beauty of Nature: Computer Explorations of Fractals, Chaos, Complex Systems, and Adaptation*. Copyright © 1998–2000 by Gary William Flake. All rights reserved. Permission granted for educational, scholarly, and personal use provided that this notice remains intact and unaltered. No part of this work may be reproduced for commercial purposes without prior written permission from the MIT Press.



(a)



(b)



(c)

Figure 10.4 Logistic map with $r = \frac{8}{10}$: (a) The time series quickly stabilizes to a period-2 limit cycle. (b) The state space of the same system shows how subsequent steps of the system get pulled into the limit cycle. (c) The state space of the same system but with only the converged values for x_t plotted, so as to clearly show the limit cycle's location.

Period Doubling Transition to Chaos

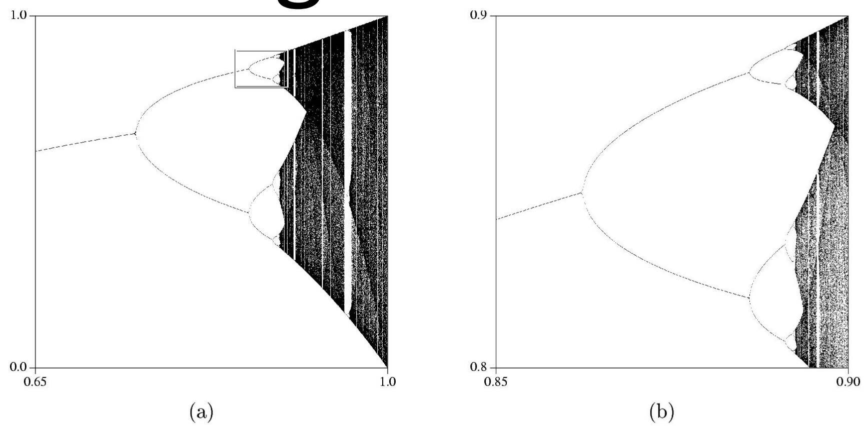
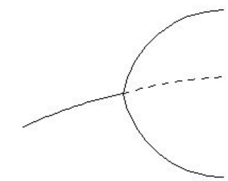


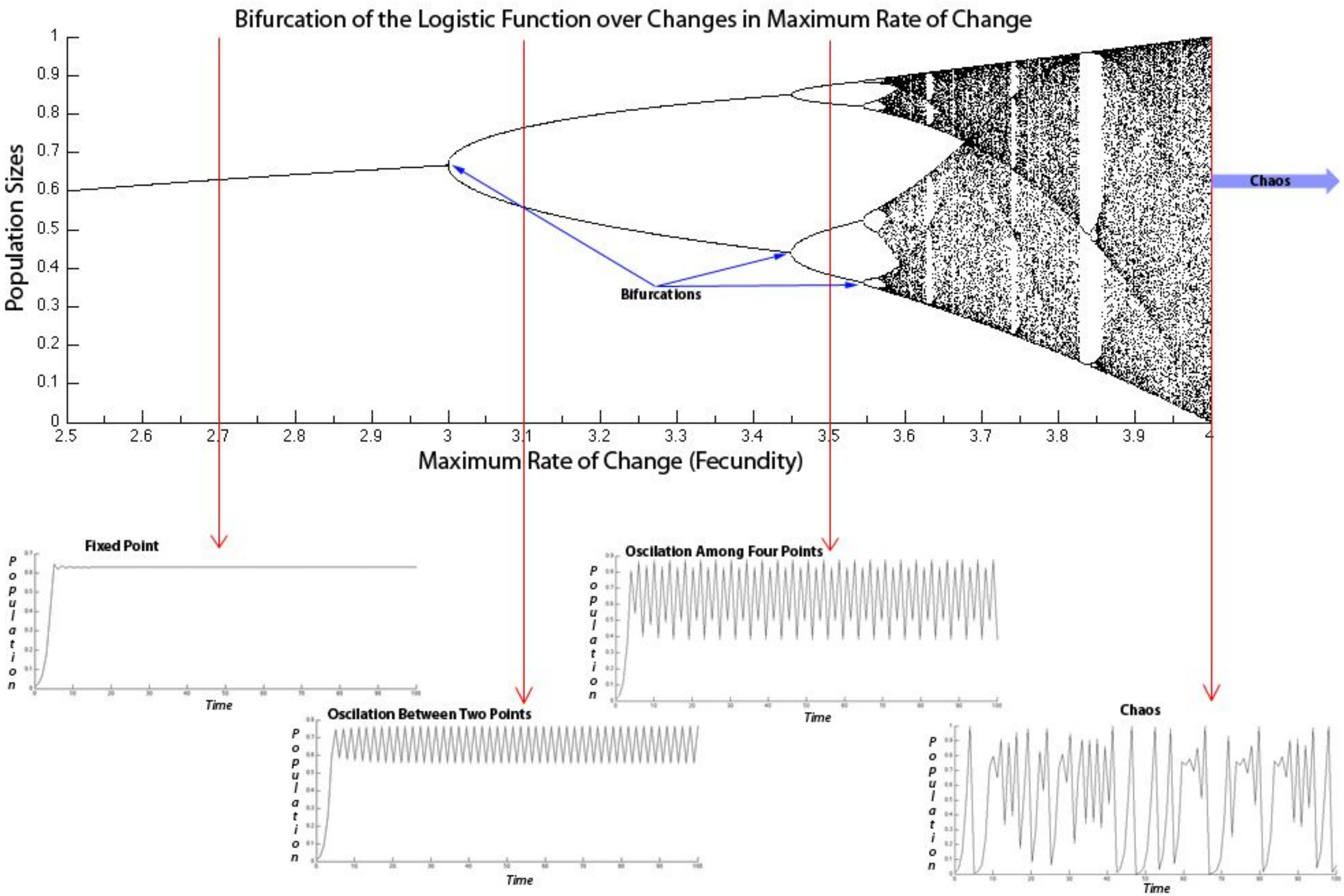
Figure 10.7 Bifurcation diagrams for the logistic map: (a) This image has values of r such that fixed points, limit cycles, and chaos are all visible. (b) This image shows the detail of the boxed section of (a).

Figure from *The Computational Beauty of Nature: Computer Explorations of Fractals, Chaos, Complex Systems, and Adaptation.* Copyright © 1998–2000 by Gary William Flake. All rights reserved. Permission granted for educational, scholarly, and personal use provided that this notice remains intact and unaltered. No part of this work may be reproduced for commercial purposes without prior written permission from the MIT Press.



X-axis: r
Y-axis: period of limit cycle

- As r increases, between ¼ and ¾:
  - For any given r, system settles into a limit cycle
  - Successive period doublings (bifurcations) as r increases
  - The amount that r increases to get to next period doubling gets smaller and smaller for each new bifurcation. (Feigenbaum’s constant)
  - At the critical value, the dynamical system falls into essentially an infinite-period limit cycle



$$x_t, r \in [0,1]$$

$$x_{t+1} = r x_t (1 - x_t)$$

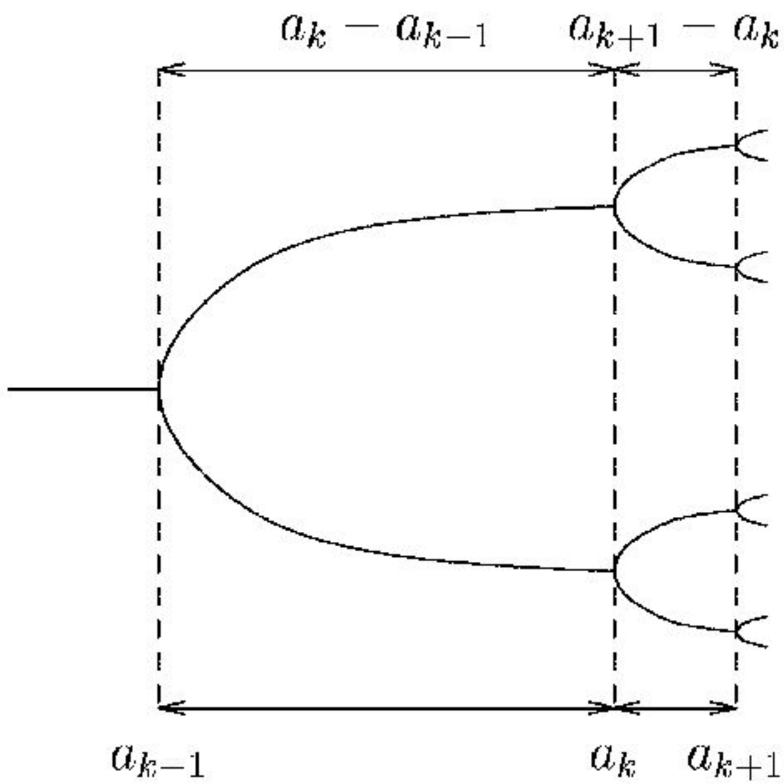


Figure 10.8 Detail of a bifurcation diagram to show the source of the Feigenbaum constant

Figure from *The Computational Beauty of Nature: Computer Explorations of Fractals, Chaos, Complex Systems, and Adaptation*. Copyright © 1998–2000 by Gary William Flake. All rights reserved. Permission granted for educational, scholarly, and personal use provided that this notice remains intact and unaltered. No part of this work may be reproduced for commercial purposes without prior written permission from the MIT Press.

Feigenbaum's constant

$R_1 \sim 3.0$ ($2^1 = 2$ period attractor)
$R_2 \sim 3.44949$ ($2^2 = 4$ period attractor)
$R_3 \sim 3.54409$ ($2^3 = 8$ period attractor)
$R_4 \sim 3.564407$
.
.
.
$R_{inf} \sim 3.569946\ldots$

The rate at which the R values converge is Feigenbaum's constant ~ 4.6692016…

How fast is the next bifurcation relative to the previous bifurcation?

From Flake: $d_k = a_k - a_{k-1} / (a_{k+1} - a_k)$
where a_k is the value of r at which the logistic map bifurcates into a 2^k limit cycle

Feigenbaum's constant is the same for all unimodal maps (with a parabolic state space) as in Mitchell Fig. 2.4 and Flake 10.2b

Characteristics of Chaos

- Deterministic
- Sensitive
- Ergodic
- Embedded

Characteristics of Chaos

- Deterministic: there is structure in the state space. No randomness. Predictable given *perfect* information.
- Sensitive: dependence on initial conditions, perturbations and numerical precision. Butterfly effect.
- Ergodic: the state space trajectory will return to all previous local regions. Life repeats itself.
- Embedded: unstable limit cycles are embedded in the chaos.

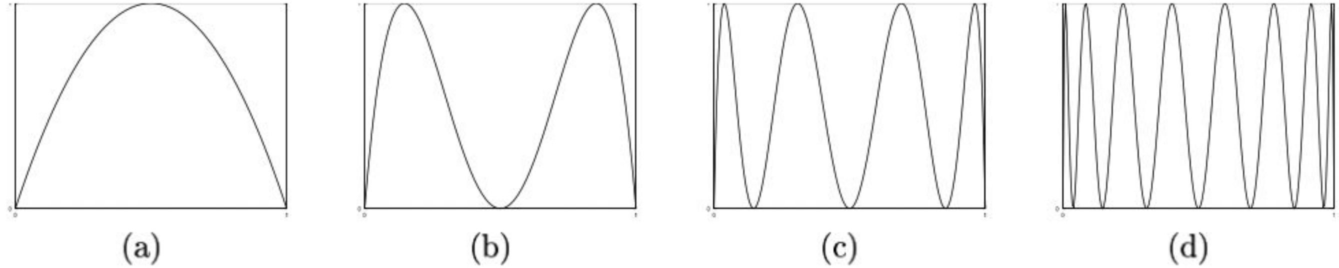


Figure 10.9 Functional mappings with $r = 1$: (a) $f^1(x)$, (b) $f^2(x)$, (c) $f^3(x)$, (d) $f^4(x)$

- Additional points from Flake:
  - Stability of fixed points
  - Pastry: stretch and fold
    - https://www.youtube.com/watch?v=iDJ6ooKHw_c
    - Short term prediction is possible, information is lost with each recursive application of the function *f(x)*
    - The floating point representation needs *m + 1* bits to predict values after *m* recursive iterations of *f*.
    - **Table 10.1: you can predict when your predictions will fail**

Logistic Map Mini Exercises

1) Work through the algebra on slides 5-6 to get the logistic map and understand what R and r are.

Given $x_{t+1} = R * x_t * (1 - x_t)$
Consider:
X_0 = 0.2 and 0.21
R = 2, 3, 3.5 and 3.9

2) Do you find: a stable fixed point, periodic cycles or chaos?
3) Where do you see sensitive dependence on initial conditions?
4) Do rabbit populations grow according to the logistic map?

   Do COVID infections?
   - What unrealistic simplifications are assumed in the logistic map?

$x_0 = 0.5$

$x_0 = 0.5000001$

$$x_{t+1} = R * x_t * (1 - x_t)$$

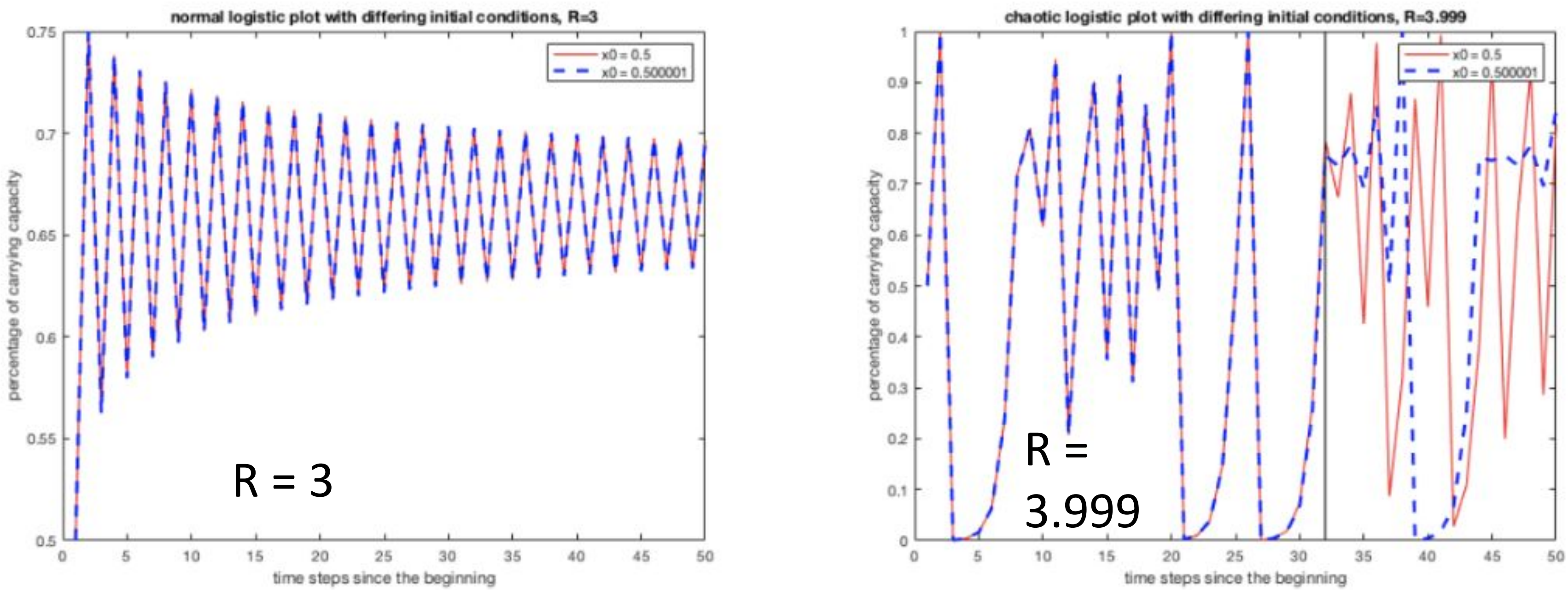


Fig. 2. Comparing SDIC on populations with periodic (left) vs chaotic (right) behavior

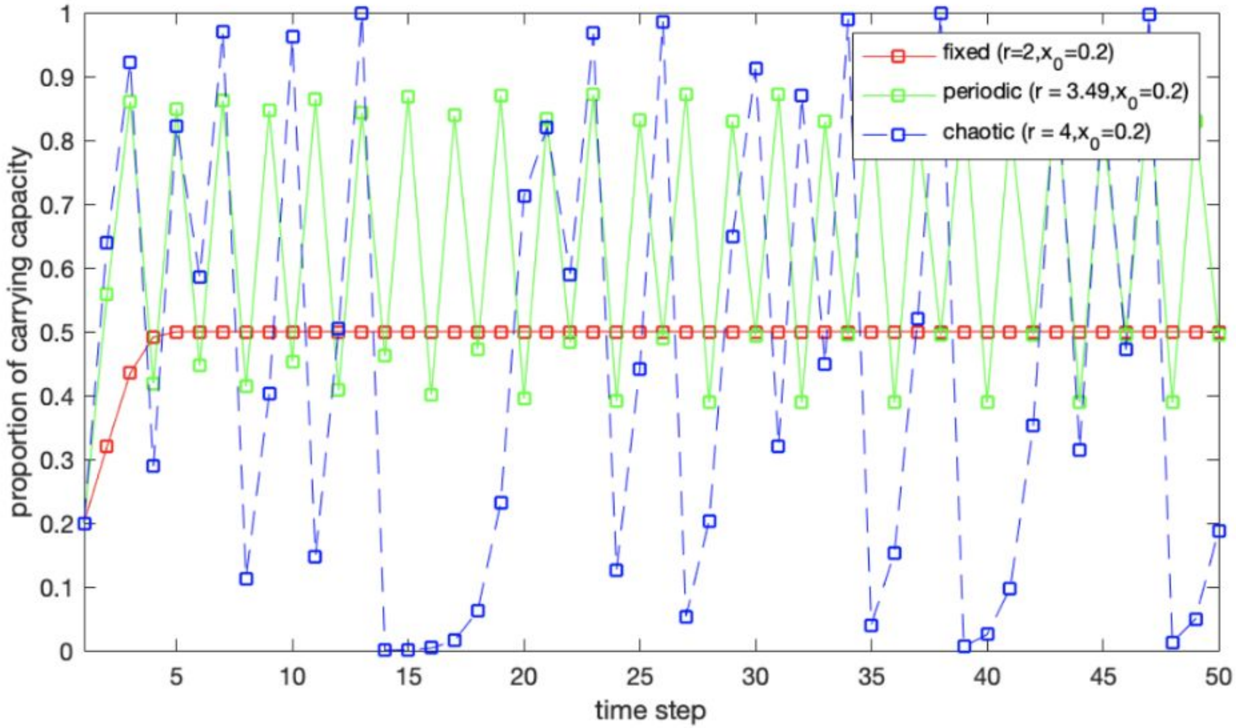


Fig. 1: Population growth for x_0=0.2 and varying values of R over 50 time steps.

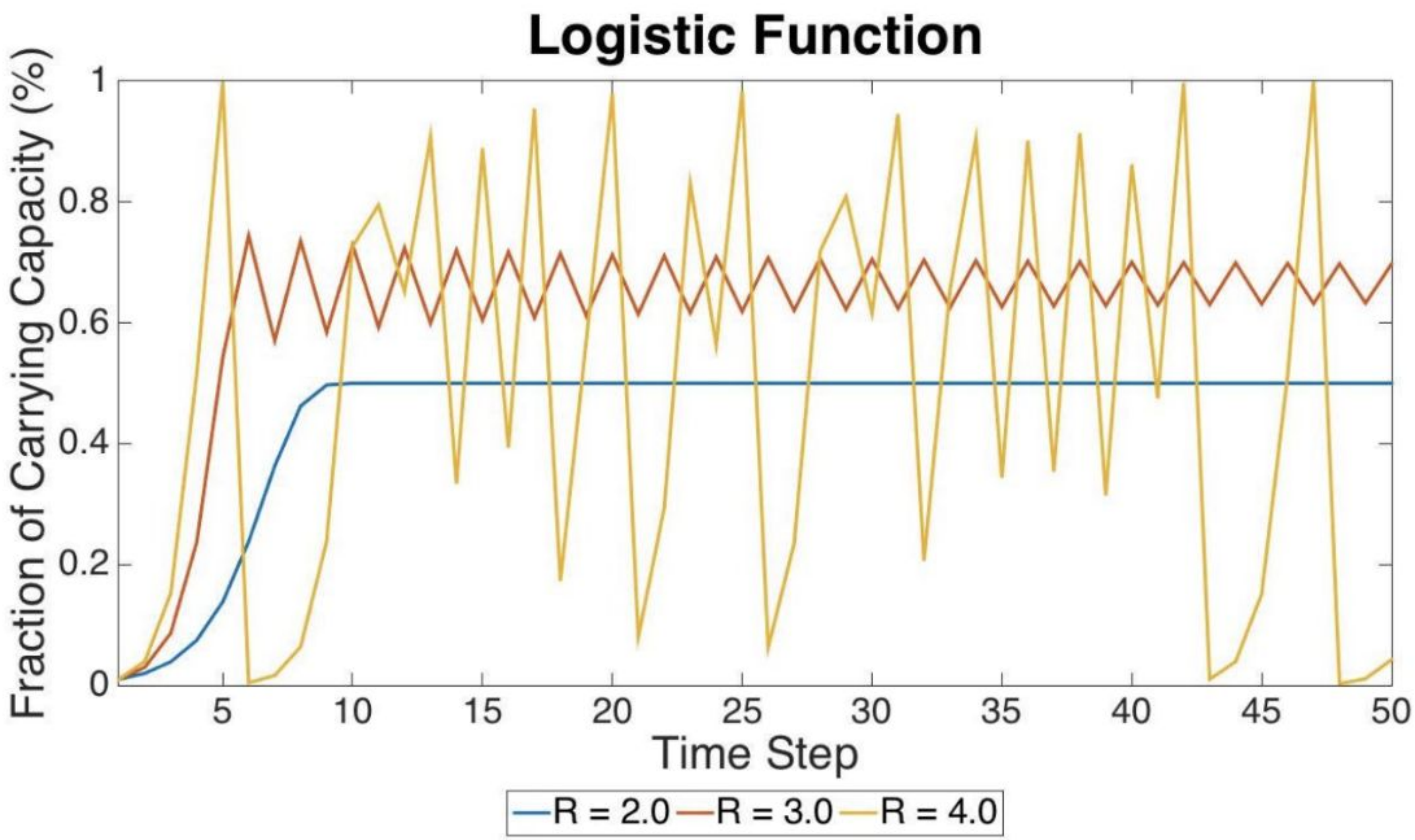


Fig. 1. Presented in this figure are three unique population trajectories generated with the logistic function. All three plots begin with a carrying capacity of 1% ($x_0 = 0.01$).

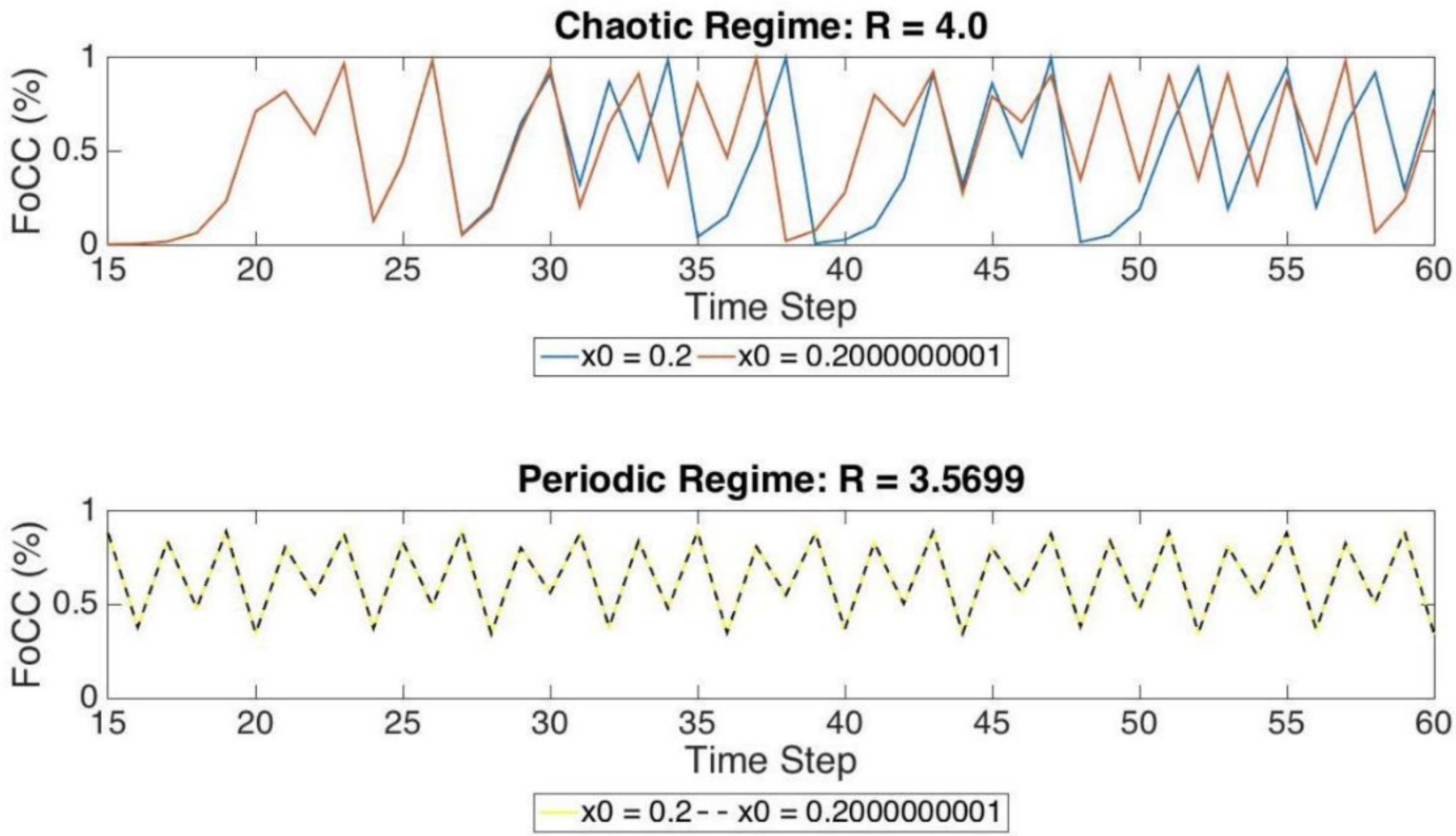


Fig. 2. Presented in this figure are four logistic function time series. The top sub-plot demonstrates the sensitive dependence on initial conditions (SDIC) for a high value of R in the chaotic regime and extremely similar values of x_0. The bottom sub-plot demonstrates that SDIC does not occur for lower values of R and extremely similar values of x_0.

Closing thoughts on the Logistic Map

- “Chaos: When the present determines the future, but the approximate present does not approximately determine the future.” - Edward Lorenz

- Logistic map: an iterated map based on difference equations with discrete values and discrete time steps. Alternatively, one can use continuous equations (and different assumptions about the dampening effects of approaching carrying capacity). Which approach is best?

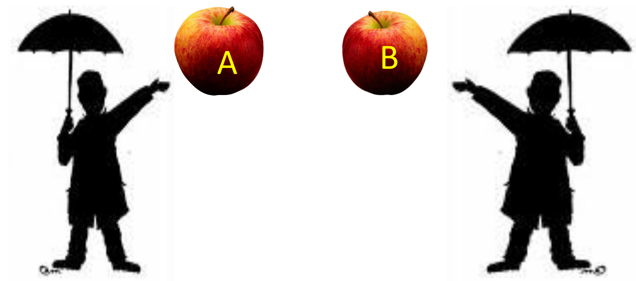
- Linear Systems can be broken down into parts. The parts can be solved independently and recombined into a solution. A great deal of practical mathematics is based on linearizing systems. A change in the initial conditions in a linear system result in a proportional change later on.

- Non-Linear Systems cannot be broken down into subproblems like linear systems. “Using a term like nonlinear science is like referring to the bulk of zoology as the study of non-elephant animals.” - Stanislaw Ulam.
- Most non-linear systems can only practically be studied through computer simulation.

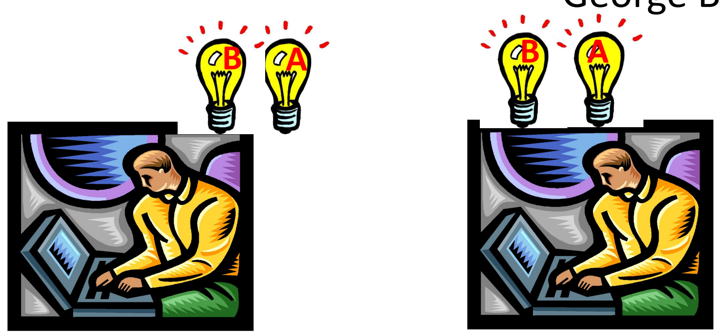
Mitchell Ch 3: Information

Information is not conserved

Information can be copied cheaply



If you have an apple and I have an apple and we exchange apples
then you and I will still each have one apple.
But if you have an idea and I have an idea and we exchange these ideas,
then each of us will have two ideas.
--George Bernard Shaw



Bateson described information as “a difference that makes a difference”

How does Mitchell describe it?

Shannon Information

- Entropy

$$H = -\sum_i p_i(\log_2 p_i)$$

- Mutual Information
- Transfer Entropy

"I blame entropy."

Shannon Information

- Shannon Entropy H to measure basic information capacity:
  - For a discrete random variable X with a probability mass function p(x), the entropy of X is defined as:

$$H(X) = -\sum p(x) \log_2 p(x)$$

  - Entropy is measured in bits.
  - H measures the average uncertainty in the random variable.
- Example 1:
  - Consider a random variable with uniform distribution over 32 outcomes.
  - To identify an outcome, we need a label that takes on 32 different values. How big does my label need to be to communicate the outcome?

$$H(X) = -\sum_{i=1}^{32} p(i) \log_2 p(i) = -\sum_{i=1}^{32} \frac{1}{32} \log_2 \frac{1}{32} =$$

Shannon Information

- Shannon Entropy H to measure basic information capacity:
  - For a random variable X with a probability mass function p(x), the entropy of X is defined as:

$$H(X) = -\sum p(x)\log_2 p(x)$$

  - Entropy is measured in bits.
  - H measures the average uncertainty in the random variable.
- Example 1:
  - Consider a random variable with uniform distribution over 32 outcomes.
  - To identify an outcome, we need a label that takes on 32 different values. How big does my label need to be to communicate the outcome?

$$H(X) = -\sum_{i=1}^{32} p(i)\log_2 p(i) = -\sum_{i=1}^{32} \frac{1}{32}\log_2 \frac{1}{32} = -32\cdot \frac{1}{32} = 5 \text{ bits}$$

00001, 00010, 00011…11111

We would like to develop a usable measure of the information we get from observing the occurrence of an event having probability p .

- We will want our *information* measure $I(p)$ to have several properties (note that along with the axiom is motivation for choosing the axiom):

1. Information is a non-negative quantity: $I(p) \geq 0$.

2. If an event has probability 1, we get no information from the occurrence of the event: $I(1) = 0$.

3. If two independent events occur (whose joint probability is the product of their individual probabilities), then the information we get from observing the events is the sum of the two informations: $I(p_1 * p_2) = I(p_1) + I(p_2)$. (This is the critical property . . .)

4. We will want our *information* measure to be a continuous (and, in fact, monotonic) function of the probability (slight changes in probability should result in slight changes in *information*).

cogs.csustan.edu/~tom/SFI-CSSS/info-theory/info-lec.pdf

Shannon Entropy Exercise

- Suppose a race has 8 horses. What is the entropy of the race outcome (how much uncertainty is there in the winner)
  - if the odds of winning are equal for all horses?
  - if the odds are: 1/4, 1/4, 1/8, 1/8, 1/16, 1/16, 1/16, 1/16?
- If your bookie tells you which horse will win, how much information would you have received (uncertainty eliminated) in each case?

- Suppose you live in Seattle where 1⁄2 the time it's sunny and 1⁄2 the time it's cloudy. I tell you whether it's sunny. How much information did I give you?

- You live in Abq, 7/8 of the days it's sunny, 1/8th it's cloudy. I tell you whether it's sunny. How much information did I give you? (Another way to ask this: how much uncertainty did I remove?)
- Do you gain more information if I tell you it's cloudy vs sunny? (trick question)

- Shannon Entropy

$$\mathrm{H}(X) = -\sum_{i=1}^{n} \mathrm{P}(x_i) \log_b \mathrm{P}(x_i)$$

- Joint Entropy

$$H(X,Y) = -\sum_{x} \sum_{y} P(x,y) \log_2 [P(x,y)]$$

- Conditional Entropy

$$\mathrm{H}(X|Y) = -\sum_{i,j} p(x_i, y_j) \log \frac{p(x_i, y_j)}{p(y_j)}$$

- Mutual Information

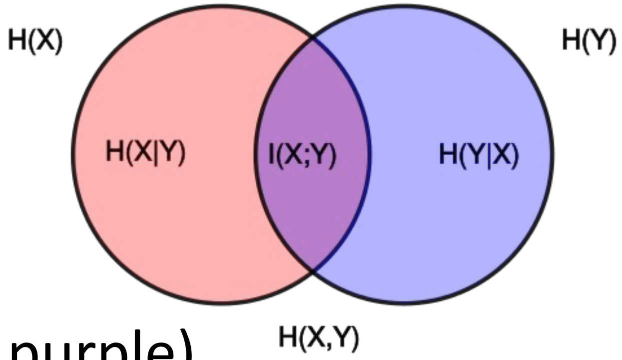
$$\begin{aligned} I(X;Y) &\equiv \mathrm{H}(X) - \mathrm{H}(X|Y) \\ &\equiv \mathrm{H}(Y) - \mathrm{H}(Y|X) \\ &\equiv \mathrm{H}(X) + \mathrm{H}(Y) - \mathrm{H}(X,Y) \end{aligned}$$

- Transfer Entropy

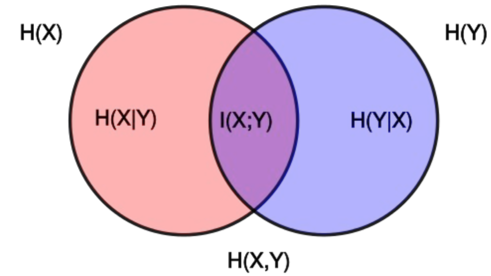
$$T_{X \to Y} = H\left(Y_t \mid Y_{t-1:t-L}\right) - H\left(Y_t \mid Y_{t-1:t-L}, X_{t-1:t-L}\right)$$

- Mutual Information

$$\begin{aligned} I(X;Y) &\equiv \mathrm{H}(X) - \mathrm{H}(X|Y) \\ &\equiv \mathrm{H}(Y) - \mathrm{H}(Y|X) \\ &\equiv \mathrm{H}(X) + \mathrm{H}(Y) - \mathrm{H}(X,Y) \end{aligned}$$



- Entropy of X: H(X), red circle (red + purple)
- Entropy of Y: H(Y), blue circle (blue + purple)
- Joint: H(X,Y): area in both circles (red, blue and purple)
- Conditional H(X|Y): red crescent
- Conditional H(Y|X): blue crescent
- Mutual Information I(X;Y): purple intersection



- Transfer Entropy

$$T_{X \to Y} = H\left(Y_t \mid Y_{t-1:t-L}\right) - H\left(Y_t \mid Y_{t-1:t-L}, X_{t-1:t-L}\right)$$

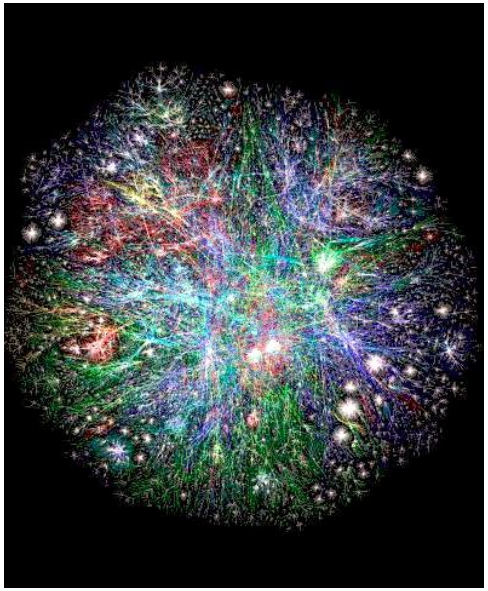
- Transfer entropy from a process *X* to another process *Y* is the amount of uncertainty reduced in future values of *Y* by knowing the past values of *X* given past values of *Y*.

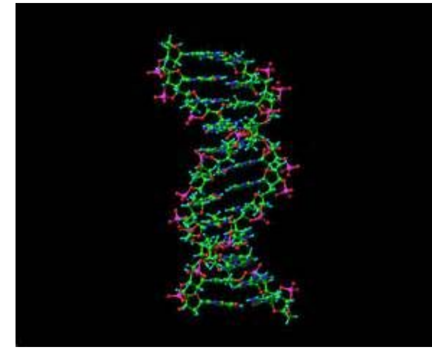
- Transfer entropy is conditional mutual information where the history of of Y (the influenced variable) is in the condition

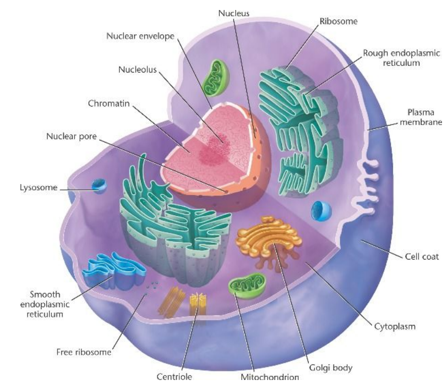
$$T_{X \to Y} = I(Y_t ; X_{t-1:t-L} \mid Y_{t-1:t-L})$$

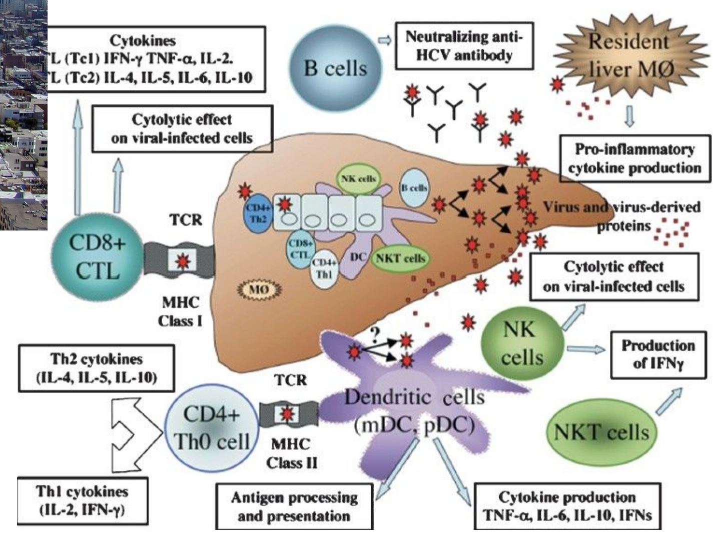
- EXERCISE: Download JIDT tool and run tutorial to calculate transfer entropy by next Tuesday

How do CAS Process Information?

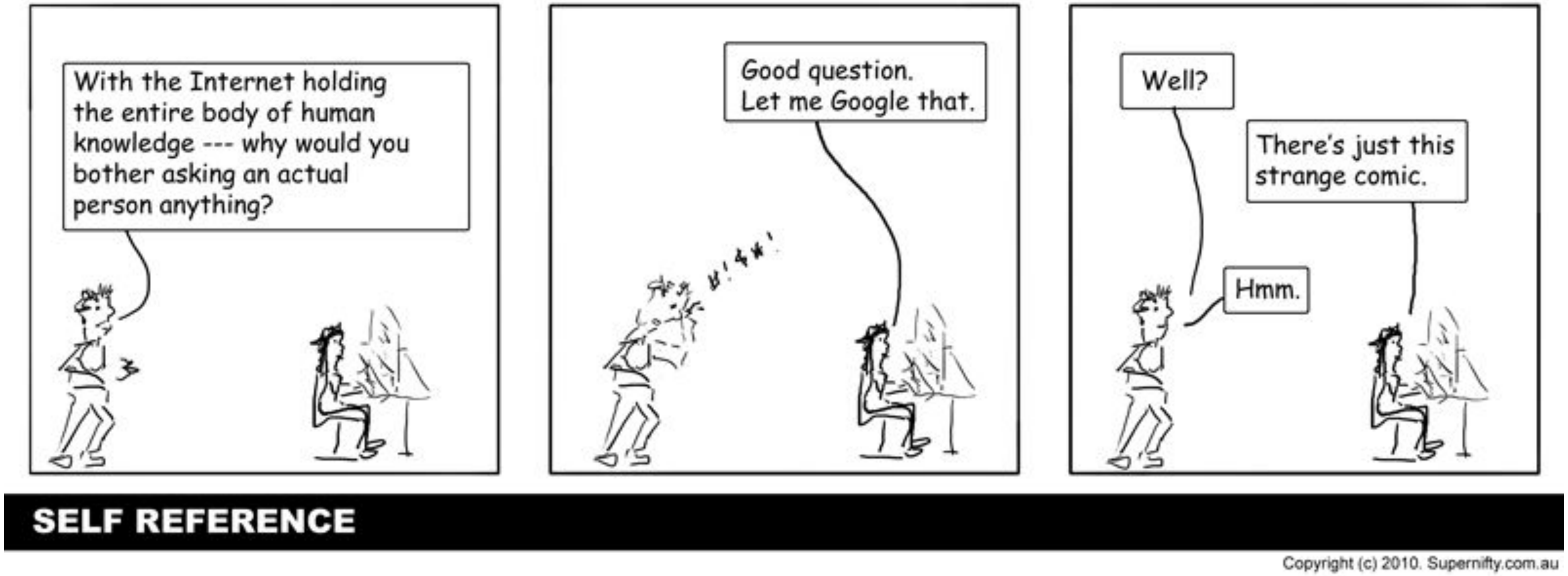








Mitchell Ch. 4
Computation and self reference



- Godel: Mathematics is *incomplete*
- Provides a mathematical proof that some true mathematical statements can’t be proved because they lead to contradictions

“This statement is false.”

- Turing: Mathematics and computation are *undecidable:* there is no ‘definite procedure’ to verify if a statement is true.
- Turing Machine is a definite procedure

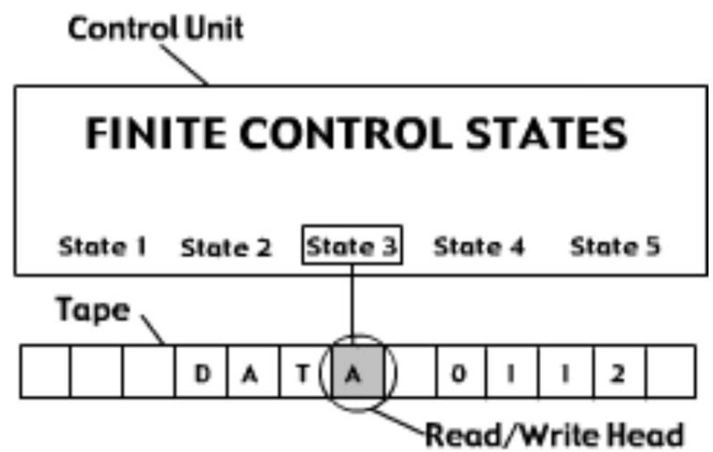
Turing Machines

Formal definition

Hopcroft and Ullman (1979, p. 148) formally define a (one-tape) Turing machine as a 7-tuple $M = \langle Q, \Gamma, b, \Sigma, \delta, q_0, F \rangle$ where

- Q is a finite set of *states*
- Γ is a finite set of the *tape alphabet/symbols*
- $b \in \Gamma$ is the *blank symbol* (the only symbol allowed to occur on the tape infinitely often at any step during the computation)
- $\Sigma \subseteq \Gamma \setminus \{b\}$ is the set of *input symbols*
- $\delta : Q \times \Gamma \rightarrow Q \times \Gamma \times \{L, R\}$ is a partial function called the *transition function*, where L is left shift, R is right shift. (A relatively uncommon variant allows "no shift", say N, as a third element of the latter set.)
- $q_0 \in Q$ is the *initial state*
- $F \subseteq Q$ is the set of *final* or *accepting states.*

1. a tape
2. A head that can r/w & move l/r
3. Instruction table
4. State register



Tape is infinite, all else is finite and discrete

Proof by Contradiction that Math/logic/computation is undecidable

- Universal Turing Machine:
  - tape contains both the input *I* and the machine (or program) *M*
- Let *I* be the *M* of a machine (specifically, the same M)
- Turing Statement (to be contradicted):

  A UTM, *H*, given input *I* and Machine *M* will halt in finite time and return Yes if *M* will halt on *I*, or No if *M* will not halt on *I*

H(*M*,*I*) = 1 if *M* halts on *I*
H(*M*,*I*) = 0 if *M* does not halt on *I*

- If H existed, it would be an infinite loop detector

- Why can't H just run *M* on *I*?

Inspiration from Godel

Create H’ that calculates H(*M,M*) except,
H’ (*M,M*) does not halt if *M* halts on *M*
H’ (M,M) halts if if M does not halt on M

- What does H’ do when it is its own input?
- H’(H’,H’) halts if H’ doesn’t halt on its own input;
- H’ doesn’t halt if H’ halts on its own input CONTRADICTION

- Godel encodes logical/mathematical statements so they talk about themselves.
- Turing encoded logic in TM and runs a TM on itself.
- The problem is
- Demonstrate that math/logic are incomplete and undecidable

Inspiration from Godel

Create H’ that calculates H(*M,M*) except,
H’ (*M,M*) does not halt if *M* halts on *M*
H’ (M,M) halts if if M does not halt on M

- What does H’ do when it is its own input?
- H’(H’,H’) halts if H’ doesn’t halt on its own input;
- H’ doesn’t halt if H’ halts on its own input CONTRADICTION

- Godel encodes logical/mathematical statements so they talk about themselves.
- Turing encoded logic in TM and runs a TM on itself.
- The problem is self-reference: “*This* statement is False”
- Demonstrate that math/logic are incomplete and undecidable