Dynamical Systems and Deep Learning: Overview

Abbas Edalat

Dynamical Systems

The notion of a **dynamical system** includes the following:

- A **phase or state space**, which may be continuous, e.g. the real line, or discrete, e.g. strings of bits 0 or 1, whose elements represent the **states** of the system.
- **Time**, which may be discrete, e.g., recursive equations, or continuous, e.g., differential or stochastic processes.
- At any given point in time, there is only one state.
- An **evolution law** that determines the state at time t from the states at all previous times.
- This defines the **orbit** or the **trajectory** of a state in the phase space.
- Interested in the long term behaviour of orbits of points.

Dynamical Systems: A simple example

- Let $\mathbb{R}$ be the state space.
- Let time be discrete $t = 0, 1, 2, \ldots$.
- $Q : \mathbb{R} \to \mathbb{R}$ with $Q(x) = x^2$ the **time independent** law:
- If at any time t the state is $x \in \mathbb{R}$, then at time $t + 1$ the state will become $Q(x) = x^2 \in \mathbb{R}$.
- At time $t = 0$ start at state $x_0 \in \mathbb{R}$ then the orbit of x_0 is:

  $$x_0, Q(x_0), Q(Q(x_0)), Q(Q(Q(x_0))), \ldots Q(Q \ldots (Q(x_0)) \ldots), \ldots$$

  also written as

  $$x_0, Q(x_0), Q^2(x_0), Q^3(x_0), \ldots, Q^n(x_0), \ldots$$

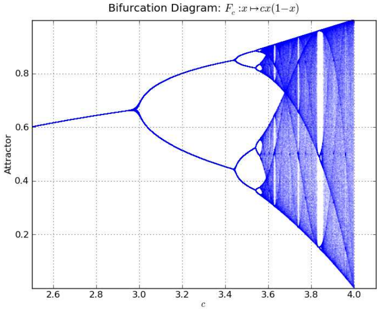
- What is the long term behaviour of such an orbit?
- If $|x_0| < 1$ then $Q^n(x_0) \to 0$ as $n \to \infty$.
- If $|x_0| > 1$ then $Q^n(x_0) \to \infty$ as $n \to \infty$.
- What happens when $|x_0| = 1$?

Basic concepts in dynamical systems

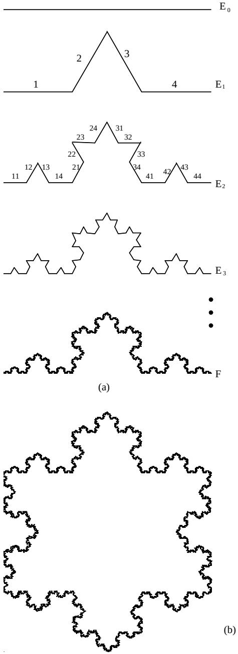
- We study **attractors, repellers, bifurcations** etc.
- Bifurcation diagram of the quadratic family

$$F_c : x \mapsto cx(1-x) : \mathbb{R} \to \mathbb{R} \quad \text{for} \quad 2.5 \leq c \leq 4.$$

- (i) Fix c and a random $x_0 \in [0,1]$.
  (ii) Plot $f_c^n(x_0)$ for $20 \leq n \leq 100$.
- For $c > 3.57$, the map F_c can exhibit chaotic dynamics: the orbit of a typical point in $[0,1]$ wanders erratically in $[0,1]$.



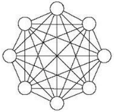
Koch curve: an example of a self-similar fractal



Agent-Based Models

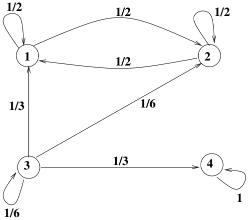
- **Agent-based models** are systems in which at any given point in time there are many interacting agents present.
- They can be considered as dynamical systems with many concurrent states.
- Agent Based Models deal directly with spatially distributed agents such as neurons, animals or autonomous agents. They can be used for learning.
- The actions and interactions of individual agents or units are taken into account with a view of assessing their effects on the system as a whole.
- We are interested to know the **emerging patterns** in the long term evolution of the interacting agents.
- These long term emerging patterns cannot be deduced using ordinary mathematical analysis applied to the local rules for the interacting agents.

A simple deterministic example: Hopfield networks



- **Hopfield networks:** the first model of associative memory in neural networks used for pattern recognition.
- N neurons with values ± 1.
- The network has a **connection or synaptic weight**, a real number, between any two neurons.
- The connection weights can be determined so as to store images in the network memory.
- State of the network is given by values of its neurons.
- There is a time independent **updating** rule that updates the values of each neuron either asynchronously or synchronously using the network synaptic weights.
- With the asynchronous updating rule the orbit of any given initial state converges to an attractor, the closest pattern in the network memory to the initial state.

A simple stochastic example: Markov chains



- **Markov chains:** Finite state space.
- E,g., a Markov chain with states labelled 1,2,3,4 as above.
- At time *t* there is a time independent probablity of transition from any state to any other state.
- We are interested in the long term behaviour of the system.
- What can be said about orbits in the above example?

Other Agent-Based Models in this course

- **Boltzmann Machines:** stochastic extension of Hopfield networks with hidden units. Learns probability distribution associated with a data set. Very inefficient.
- **Restricted Boltzmann Machines (RBM):** have connections only between any hidden unit and any visible unit. A simple training algorithm has revolutionised machine learning,
- **Deep Belief Nets:** are obtained by stacking RBM's. Consistently outperformed many rival techniques.
- **Small World Networks:** which model social and biological networks, are distinguished by low average path length, high clustering and scale-free properties.
- **Kaufmann Networks:** are Boolean networks which model gene mutation and evolution.

Computational complexity: Big O Notation

- Let $f(x)$ and $g(x)$ be two real-valued functions defined on some subset of $\mathbb{R}$ (e.g., $\mathbb{N}$). One writes

$$f(x) = O(g(x)) \quad \text{or} \quad f(x) \in O(g(x))$$

  as $x \to \infty$ if for sufficiently large values of x, the value $f(x)$ is at most a constant times $g(x)$ in absolute value.
- That is, $f(x) = O(g(x))$ if there exists a positive real number M and a real number x_0 such that $|f(x)| \leq M|g(x)|$ for all $x > x_0$.
- **Examples**:
  - $-2x^3 \log x + x^2(\log x)^4 - 4x = O(x^3 \log x)$.
  - $x^4 + 6 \times 2^{x+7} - 3^{x-13} = O(3^x)$.

Little o Notation and Equivalence $\sim$

- Given a real-valued function f defined on some subset of $\mathbb{R}$ and $a \in \mathbb{R}$, we say $f(x) \to a$ as $x \to \infty$ if for any $\epsilon > 0$ there exists $K > 0$ such that $|f(x) - a| < \epsilon$ for all $x > K$.
- Let $f(x)$ and $g(x)$ be two functions defined on some subset of the real numbers. One writes

$$f(x) = o(g(x)) \text{ or } f(x) \in o(g(x))$$

  as $x \to \infty$ if $f(x)/g(x) \to 0$ as $x \to \infty$.
- So, in words, $f(x) = o(g(x))$ if $f(x)$ is negligible compared to $g(x)$ for large enough x.
- We write $f \sim g$ (i.e., f and g are equivalent) as $x \to \infty$ if $f(x)/g(x) \to 1$ as $x \to \infty$.

Examples

- $(\log x)^n = o(x^a)$ as $x \to \infty$ for any $n > 0$ and any $a > 0$.
- $P(x) = o(2^x)$ for any polynomial P as $x \to \infty$.
- $\sin(1/x) \sim 1/x$ as $x \to \infty$.
- $-3x^4 + 10x^3 + 8x^2 \sim -3x^4$ as $x \to \infty$.
- $$\frac{x^2 + 7 \times 2^x}{7x^9 + 3^x - 5^x} \sim -7(2/5)^x \text{ as } x \to \infty.$$

Asymptotic behaviour

- We can use the Big O notation to describe the **space complexity** (how the CPU or memory resources vary with the algorithm's input size) as well as **time complexity** (how the time taken for the algorithm to complete varies with its input size).
- We may be interested in the **best**, **worst**, and **average cases**. By default it usually refers to the average case, using random data.
- The frequently encountered O values are: constant $O(1)$, logarithmic $O(\log n)$, linear $O(n)$, $O(n \log n)$, quadratic $O(n^2)$, cubic $O(n^3)$, polynomial $O(n^d)$ for some $d \in \mathbb{N}$.
- We also use the $\sim$ notation to describe the **asymptotic behaviour** of characteristic quantities in dynamical and complex systems.

Organisation

This course consists of

- **20** lectures;
- **8** tutorials;
- **2** assessed courseworks;
- Paragraphs, pages or subsections or exercises that are labelled with (*) are non-examinable although they are useful to know to follow the course.
- Some of the pictures in the notes have been reproduced from the books listed as references.

Suggested Reading

- Murphy, K. P., Machine learning: A probabilistic Perspective. MIT Press 2012
- Hinton, G., A Practical Guide to Training Restricted Boltzmann Machines, 2010, Available on-line.
- Devaney, R. L. *An Introduction to Chaotic Dynamical Systems.* Westview Press, 2003.
- Gros, C. *Complex and Adaptive Dynamical systems.* Springer, 2008.
- Hertz, J. and Krogh, A. and Palmer, R.G. *Introduction to the Theory of Neural Computation.* Addison-Wesley, 1991.