

Computational Biology: An Overview

“The book of nature is written in the language of mathematics...”

- Galileo

Traditionally, mathematics has been an integral part of physical and engineering sciences, but until recently biology has been excluded from its purview. This is rather unfortunate, as in the last couple of decades the importance of mathematics and computers in biology has steadily been growing. So much so, that math and computers have now become an integral part of biological sciences as much as physical and engineering sciences. Some of the following developments seem to have triggered this change:

- 1) The exponential increase in the biological data, including DNA and protein sequences, gene expression data, DNA and protein structures, has spurred the development of computational tools to store, search, retrieve and analyse the data.
- 2) The study of simple nonlinear dynamical systems which exhibit a range of dynamical behaviour, including steady state, periodic orbits and chaos, where deterministic equations lead to seemingly random trajectory, raised the hopes of modelling complex biological behaviour.
- 3) Game theory, where one tries to predict the outcome of a game given a number of players (≥ 2) according to the strategy each player adopts, has been used to model behaviour in biological systems.

This lecture presents the kind of problems in biology that can be addressed using computational tools and the tools that are widely used. Each of these topics can become a one semester course, so we will briefly touch upon some of the topics. I don't claim the lecture to be comprehensive, so if some topics are left out it will mostly be due to time constraints. The approach I will take is to first present a broad list of biological problems where computational methods can give an insight. Then I will list out the different kinds of algorithms and tools which are used to tackle the different problems. This will be followed by a few examples where these tools have been used successfully.

Let us start with a list of broad areas in biology where computational tools have been used with some success.

- 1) Pattern recognition: Detecting patterns from incomplete information. For example, finding genes in genomic sequence, looking for pattern of gene expression in microarray data, searching conserved motifs in protein, DNA or RNA sequences, finding repeats in DNA/protein sequences, recognising secondary structure in protein and in DNA or RNA, classifying protein structures, predicting active or functional sites in proteins to name a few. The examples quoted here have the partial information available in the form of DNA/protein/RNA sequence and we are trying to infer some kind of function from them. So, in a sense pattern recognition tries to extract hidden information.
- 2) Pattern formation and characterisation: Here the aim is to understand what kind of dynamics could have led to the patterns observed in nature. Some of the well known examples of natural patterns are those seen on animal coats, like the stripes on a zebra, spots on cheetah and giraffe, intricate patterns on seashells, in leaves

- and flowers, etc. Obviously pattern recognition problems are different from pattern formation and characterisation problems. When studying pattern formation the aim is to simulate the dynamics that leads to an observed pattern, for example stripes on zebra or spots on a cheetah, whereas pattern recognition deals with problem of detecting hidden patterns.
- 3) Structural modelling involves modelling the dynamics of bio-molecules leading to an understanding of how proteins fold, how different bio-molecules interact with each other, how cellular membranes behave under different conditions etc. It also includes energy minimisation techniques of protein structures predicted by homology modelling.
 - 4) Modelling of macro-systems spans a very wide area of research, including what is now becoming popular as systems biology. It covers modelling of gene networks (as in biochemical pathways), neural network, dynamics of organs (such as heart, brain), populations, ecosystems (such as interacting species) etc.
 - 5) With the advent of the microarray and proteomics techniques, image processing has become an integral part of biological experiments. It involves identifying the centre of the light emitting dots correctly, removing background intensity (which is usually not uniform across the array), extracting ratio of intensities when two different colour probes are used, comparison of slides from different arrays (important when looking at temporal expression of a set of genes) etc.
 - 6) Data management and warehousing have gained importance not only due to the exponential increase in the size of data, but also because of the different formats in which biological data is present, including simple text, tables, images etc. To be able to extract relevant information from the raw databases, statistical analysis can be used as sift to filter out interesting data.

We have gone through a representative list of fields in biology that require assistance from the mathematician and the computer scientist. We will now run through a list of computational tools that are widely used in many of the applications designed for analysing biological system.

- 1) Dynamic programming (DP) algorithm is used in optimisation problems, where minimisation or maximisation of some measure is required. DP can be applied to problems where the state of a system at time point t (space point x) is dictated by the state of the system at time point $t-1$ (space point $x-1$), and the final state holds the optimal value. The most popular use of this tool in bioinformatics is in the pair-wise alignment of two sequences, where optimisation of the score of the aligned sequences is required.
- 2) Markov model, artificial neural networks and Fourier transform are well suited for pattern or motif recognition. Markov models are used to define transition probabilities from one state of the system to another, given a finite number of states the system can adopt. For example, in gene finding the observed states are promoter, transcription start site, 5' UTR, translation start site, exon, intron, termination site, 3' UTR, poly-A signal. Markov model can be used to compute transition probabilities between the states and predict the gene. In a hidden Markov model some states are hidden. In the example of gene finding, gene is the hidden state, while the others are observed states. Artificial neural networks are

designed to loosely model the functioning of the brain. It is made of several layers, each layer comprises of a set simple processing units called neurons. Each neuron may be linked to one or several neighbours with varying coefficients of connectivity. Learning is achieved by adjusting these coupling strengths. There is an input layer, followed by one or more layers for processing the information and an output layer giving the results. The network works by a feedback mechanism, where the output of the network is compared with the desired output and the connectivity strengths are adjusted accordingly. Neural networks are used in a variety of different problems, but are best known for their pattern recognition and classification capabilities. Fourier transform is statistical tool used to unravel the correlation structure between different time or space points. It can be used in pattern recognition problems.

- 3) The third set of tools again optimisation techniques mostly used in simulations of some dynamics, for example the dynamics of protein folding. Molecular dynamics simulation is deterministic method of searching the phase space of a dynamical system to find the global minimum or maximum. On the other hand, Monte Carlo and Genetic Algorithm are stochastic methods to achieve the same goal. In Monte Carlo approach you randomly jump from one state of the system to another, if it lowers the energy or with some probability if the energy increases. Genetic Algorithms are artificial life systems in the sense that they simulate the evolutionary mechanisms of mutation and recombination (crossover) to arrive at the optimal solution. A fitness function is used to evaluate the fitness of the offspring, and reproductive success varies with fitness.
- 4) Cellular automata algorithms are designed to study dynamics which is discrete both in time and space. An example of dynamics discrete is space could be the growth of a population of species whose generations do not overlap. Many insects like the butterfly follow this pattern. The most famous example is Conway's game of life.
- 5) Game theory has been used to predict behaviour of interacting species by considering different strategies each individual adopts.
- 6) There are a host of tools for doing statistical analysis, including clustering methods, assessing the significance of association of a genotype to a phenotype, assessing significance of aligned sequences etc.

Having introduced the field in its widest coverage, we will now concretise the ideas by quickly running through a few examples where we will see how these tools have been applied and what has been achieved in the process. The first example will exploit the development of chaos theory, so before presenting the paper I will give a brief introduction to chaos theory.

Deterministic chaos is one of the many dynamical behaviours of a non-linear system. It was first observed by Edward Lorenz in 1960 when he was studying weather dynamics using simplified hydrodynamic equations on a primitive computer. He observed certain patterns in the numbers generated by the dynamics, which he could use to predict what would happen next. While trying recreate one of the patterns he had observed earlier, by typing in the numbers printed out by the computer, he found that the pattern produced

quickly diverged from the original pattern, and was quite different at the end of a few time steps. After much analysis of these results, the only reason for this discrepancy was attributed to the fact that the numbers printed had only 3 decimal places (for example 0.617), whereas the computer used numbers to 6 decimal places (for example 0.617395) in its computation. This small error grew quickly, so that the dynamical pattern was different in a few time steps. This is a typical feature of chaotic dynamics. Small errors/perturbations grow exponentially fast. Metaphorically, the flapping of the wings of a butterfly in Tokyo can raise a storm in New York. This property makes the trajectory of a chaotic system resemble that of a random process, even though the dynamics is governed by a set of deterministic equations, therefore the name deterministic chaos. You may ask then, how can we distinguish a chaotic trajectory from a random one. One way to determine this is to construct the first return map. The first return map is the plot of x_{t+T} vs x_t . The return map of a periodic dynamics is a finite set of points, that of random dynamics is a scatter plot covering the entire space, whereas for a chaotic system we get a scatter plot on some geometric structure (for example a parabola).

Chaotic dynamics is just one a host of behaviours exhibited by a complex dynamical system. Complex dynamical systems are usually governed by a set of coupled non-linear equations. They exhibit a range of dynamical behaviours dependent on a set of parameters of the system, including stationary state, periodic behaviour and aperiodic or chaotic dynamics. The simplest of systems exhibiting complex dynamical behaviour is the logistic map: $x_{n+1} = \mu x_n(1 - x_n)$. It describes the growth of the population of a species whose generations do not overlap. This simple system exhibits a whole range of behaviour as a function of the parameter μ . The most interesting behaviour is observed for the variable x lying between 0 and 1 and the parameter values ranging from 0-4. For μ values 0-3 the dynamics is quite boring as the system reaches a stationary state in equilibrium. The only interesting thing that happens is at $\mu = 1$ where the stable equilibrium value changes from 0 to $(1 - 1/\mu)$. All the interesting dynamics is observed for parameter values 3-4. We can summarise this behaviour by plotting the asymptotic dynamics as a function of the parameter value, known as the phase plot. Fig.1 describes the phase plot for the logistic map. As can be clearly seen in the figure at $\mu = 3$ there is a transition in the dynamical behaviour, the stationary state disappears and gives rise to a 2-period trajectory. What really happens is that the stationary state becomes unstable and the 2-period orbit emerges as the stable state. This kind of transition is known as the pitchfork bifurcation. A little beyond $\mu = 3.4$ the 2-period dynamics becomes unstable giving rise to a 4-period. This continues until we get aperiodic or chaotic trajectories. This is known as the period-doubling route to chaos. We see then that in the chaotic regime a large number of unstable periodic orbits co-exist.

Another simple system is the Bernoulli map: $x_{n+1} = 2x_n \text{ mod } 1$, where $0 \leq x \leq 1$. For example, take $x = 0.2$. The sequence of numbers is: 0.2, 0.4, 0.8, 0.6, 0.2, 0.4, 0.8, 0.6, 0.2, ... As is clear this is a 4-period trajectory $x_{n+4} = x_n$. Now let us change the initial value to $x = 0.21$. Now the trajectory reads: 0.21, 0.42, 0.84, 0.68, 0.36, 0.72, 0.44, ... Again, we see a small change growing very fast and leading to totally different trajectory

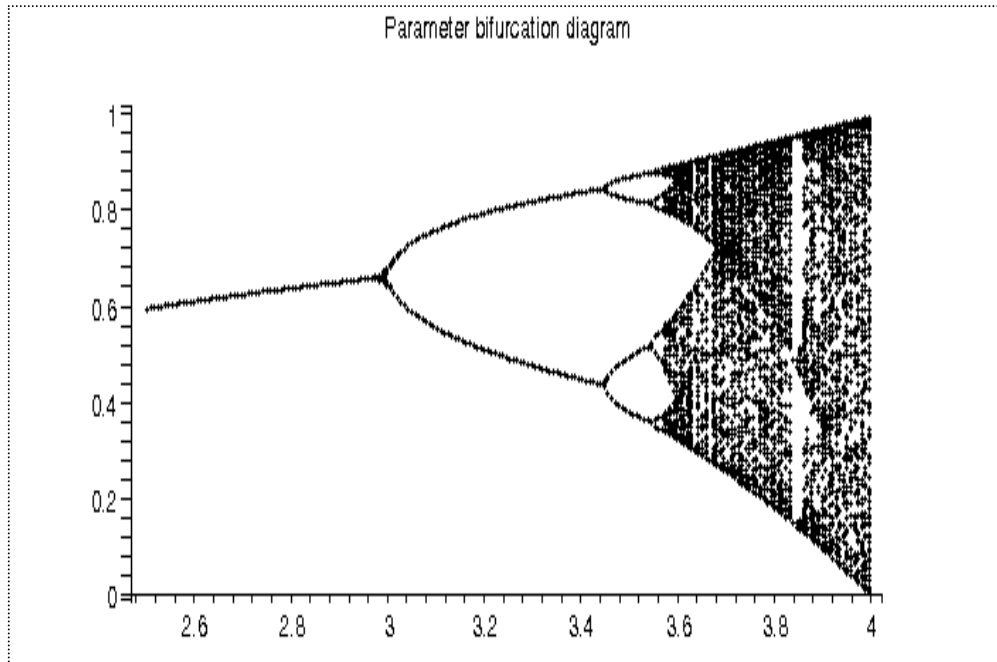


Fig.1

after a few time steps. An intuitive way of seeing how this happens is to convert these numbers into binary mode. When we do that, $x = 0.2$ now becomes $x = 0.001100110011\dots$ whereas $x = 0.21$ becomes $x = 0.001101011100\dots$. Now, the operation of doubling and removing the integer part is equivalent to shifting the decimal point to the right by one place and removing any 1's that come to the left of the decimal. You can verify yourself that for $x = 0.2$, we get a 4-period trajectory because in the binary format the number is a repetition of "0011". We also see that 0.21 is different from 0.2 at the 6th place in the binary format, and that is why the trajectory becomes completely different after the 6th time step.

This feature of a chaotic system is known as extreme sensitivity to initial conditions and it is responsible for making chaotic dynamics unpredictable. The beauty of chaos is that the very property which makes it unpredictable makes it amenable to control, thus even though we cannot predict chaos, we can control it. There are several kinds of controls including suppression of chaos, enhancing chaos, stabilisation of an unstable periodic orbit and synchronisation of chaos. In some instances, for example in lasers and some electronic devices, it is necessary to suppress chaos to improve their efficiency, whereas maintaining or enhancing chaos helps fluids mix better, and this is useful to improve combustion in for example vehicular engines. There are yet other applications where staying in the chaotic regime one forces the system to maintain one of the several unstable periodic orbits present. For example, when applied to a laser system, this kind of control allows us to tune the laser to several different frequencies of emission.

An application of chaos control in biology appeared in Science where cardiac arrhythmia induced in a rabbit was controlled using small electric pulses. Arrhythmia was introduced in a rabbit heart by injecting it with the drug ouabain. The first return map of the inter-

beat interval was traced and unstable periodic orbits with saddle instability were identified. Small pre-computed electrical impulses were applied at appropriate intervals to bring and maintain the inter-beat interval in the selected periodic motion. A later study applied a similar approach to the brain to control epileptic seizures. A major advantage of this approach to control cardiac arrhythmia is that instead of giving massive painful shocks to the patient, small electrical pulses are used to achieve the same therapy.

The second example I will present involves game theory, so before presenting it I will briefly explain in a very simplified manner what game theory is. We have all played some game or the other in our life, so the concepts should be easy to grasp. In a game we require two or more players (of course there are solitaire games also, but we are taking the more generic case). A game has one or more outcomes, win or lose (or tie), survive or die, etc. The outcome of a game depends on the strategy each player adopts. So, for example if we consider driving a vehicle in the traffic a game, it is not enough that you practise safe driving to avoid accidents. Everybody on the road has to practise safe driving to make a commute accident free. A popular example of a game is the prisoner's dilemma. Two men have been arrested for a crime and are kept in separate prisons. Each prisoner has two options: 1) to confess, 2) not confess. There are three possible outcomes of the game: 1) one prisoner confesses the other does not, 2) both prisoners confess, 3) both prisoners do not confess. The pay-offs determine the strategy a prisoner will adopt, and in this case they are defined as the number of years in prison. In the first case, the prisoner who confesses is freed and the other serves 15 years in the prison. In the second case both get 5 years and in the third case both get 1 year.

Now that you have a general idea how game theory works, we will go through an example of a biological system where it was applied. The aim of the work I am going to describe was to see the effect of predator behaviour on the prey-predator dynamics. The interaction of prey with predator is described by the classical Lotka-Volterra model. The predators can adopt one of the following two behaviours: 1) a predator can fight another predator to keep the prey it has captured (hawk strategy), or 2) or it can scam and let the other predator keep the prey (dove strategy). The following are the assumptions of the model:

- 1) Gain depends on the prey density, which modifies predator behaviour
- 2) The prey-predator interaction acts at a slow time scale
- 3) The behavioural change of predator works on fast time scale
- 4) At a given time a predator with a prey captured is challenged by only one predator

The first assumption seems obvious. For example, when prey is abundant we may expect more of dove behaviour, whereas hawk behaviour is expected when prey is scarce. The following two assumptions imply that once a predator has captured a prey, it may have to fight more than one predator to keep the prey. However, according to the last assumption there is always a one-to-one fight, *i.e.* at any time a predator can be challenged by only one predator. The study was carried out for different prey densities. The startling conclusion of the study was that hawk behaviour is prevalent when prey density is high, while at low prey densities there is a mix of hawk-dove behaviour. This is contrary to the intuitive assumption made at the beginning. This leads to a change of view, namely it is

not profitable to be aggressive when prey density is low and it is better to cooperate, whereas when prey density is high one can try to monopolise the prey.

In this lecture I have tried to give you a flavour of the field of computational biology. Many areas were not covered, and even those that got mentioned were not dealt with in detail. But I hope this was enough to arouse your curiosity and inspire you to go back and read more about the subject.