

Fitting Genome Models To Known Virus Structures

Project Description

by Saskia de Groot & Jotun Hein

The explosion of available viral genomic data in recent years has provided us with a new platform of analysis. There are currently 2731 genomic reference sequences for 1782 different viral genomes (see [GenBank, 1]), and numbers are rising. Many papers have been published looking at nucleotide composition of certain regions in specific genomes, or attempting to describe the evolutionary behaviour of overlapping reading frames. These are all exciting questions, but maybe it would be worthwhile to also approach the problem the other way round? As opposed to intricately studying one genome in isolation in order to understand viral genomes in general, we wish to draw large-scale conclusions by analyzing the whole of the Genbank viral dataset in one go. We could find out the average nucleotide composition of viruses, how likely a start codon is a true start codon, what the probability of an overlapping reading frame opening in an already existent coding region is, et cetera. But how would we go about this?

For a long time people have been using what is known as Hidden Markov Models (HMMs) to model genomic data. An HMM consists of four things: a set of states S , an alphabet C , a set of state transition probabilities T and a set of emission probabilities E . Each state s emits a character c with probability $e_s(c)$. Transitions between two states s_i and s_j are made with probability t_{ij} . Take, for example, the case of a casino playing with a fair and a loaded die, where the loaded die has probability $1/2$ of rolling a 6 and $1/10$ of rolling any of the other five numbers. After each throw we stick to our die with probability 0.9 and switch dice with probability 0.1 . The set of states here is $S = \{fair, loaded\}$, our alphabet is $C = \{1, 2, 3, 4, 5, 6\}$, our set of transition probabilities is $T = \{0.9, 0.1\}$ and the set of emission probabilities is $E = \{E_{fair}, E_{loaded}\}$ with Suppose a sequence of throws is

$$\begin{aligned} E_{fair}(x) &= 1/6 & \text{for } x \in 1\dots 6 \\ E_{loaded}(x) &= 1/2 & \text{for } x = 6 \\ &= 1/10 & \text{for } x \neq 6 \end{aligned}$$

made from one die or the other, then the state — i.e. *fair* or *loaded* — is hidden, hence Hidden Markov Model. Knowing the architecture though, if we see 1, 2, 4, 6, 3, 1, 5, 6, 6, 6, 6, 5, 6, 6, 3, 5, 4, 1 as a sequence of throws we can still make a good guess about which bits have been thrown with the loaded die and which with the fair one. There are many famous algorithms, such

as the Viterbi and the Forward-Backward algorithm which provide computationally feasible ways of finding a good solution to this and other related problems [Durbin *et al.*, 1998].

When devising models for genomic data, we can attempt to model several things. For example we could have one state which is *coding* and one state which is *non-coding*. Our alphabet could be the four nucleic acids A, C, G, T and we let the probabilities of emitting each one be dependent on the prior two, thus accounting for the three-periodicity within a coding region. These probabilities are thus given by a $2 \times 4 \times 4 \times 4$ matrix P . Our state-transition probability from non-coding to coding will be α if we see a start codon, and from coding to non-coding it will be 1 if we see a stop codon. If we do not know which regions of the genome are coding or non-coding, but we do know the P -matrix, we may use this to find the coding regions. If however, we already know the coding and non-coding regions, we may use this information to devise a matrix P that fits the data best.

On a more complicated note, imagine an HMM which models nucleotide composition equivalent to the above, but also draws overlapping reading frames into account [McCauley & Hein, 2006] — a common feature in viral genomes. This means that we now have to distinguish not just between coding and non-coding regions, but indeed between non-coding, single coding, double coding and even triple coding regions. The emission matrices within these regions are all going to be different, due to the nucleotide dependencies at each position. We now may introduce different state transition probabilities such as α_{01} , α_{12} , α_{23} for the transition from non to single to double to triple respectively (see figure 1). Again we can estimate both the emission and the transition probabilities from the data once we know the gene annotation.

Using these Hidden Markov Models, but making them non-hidden by feeding them the gene annotation off Genbank, we will be able to estimate an array of parameters describing genomic behaviour. We could look at parameters on a global scale, or compare them between different organisms to discover certain trends and tendencies. This is a very exciting and promising project — it gives excellent insight into the methods and questions prevalent in the bioinformatic field and a simple idea could lead to very strong results! A simple project plan could work along the following line

- Read up on Hidden Markov Models in general.
- Consider various specific models which apply to gene annotation, in particular viruses.
- Evaluate what sort of genomic features, i.e. nucleotide composition, frequency of start and stop codons, length of introns etc. we would like to investigate.
- Devise models to accommodate for said parameters.

- Fit the above models to GenBank to estimate parameters and evaluate results.

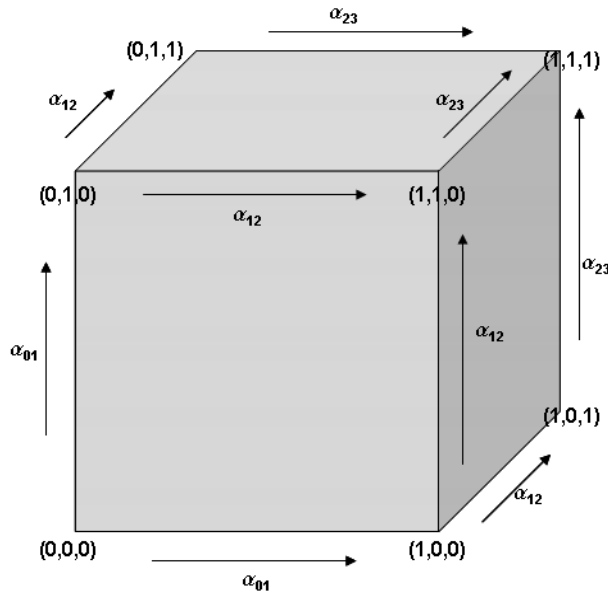


Figure 1: The cube representing the overlapping reading frame model. Since each codon consists of three nucleotides, there are three “global reading frames”, known as gRF’s. Here (0,0,0) refers to non-coding, (1,0,0) refers to single coding in gRF1, (1,0,1) refers to double coding in gRF1 and gRF3 and so on. The transition probabilities between the different states are shown, and the representation on the cube demonstrates which transitions are allowed, and which are not.

References

- [Durbin *et al.*, 1998] Durbin,R., Eddy,S., Krogh,A., Mitchison,G. (1998) Biological Sequence Analysis, *Cambridge University Press*.
- [GenBank, 1] All viral data is publicly released on the GenBank database, see <http://www.ncbi.nlm.nih.gov/>
- [McCauley & Hein, 2006] McCauley,S., Hein,J. (2006) Using HMMs and observed evolution to annotate viral genomes, *Bioinformatics*, Advance Access published online on April 13, 2006