Sequences, Genes, and HMMs

Mark Garnett — Student ID 25969722

1 Question 1

The first codon in an ORF must be ATG and the final codon one of TAA, TGA, and TAG, with L-2 codons in between which cannot be a stop codon. The probability of the start codon at position 1 is 1/64, and the final codon at position L is 3/64. The probability of any non-stop codon is 61/64. Thus, the probability of the ORF is:

$$p(ORF) = \frac{1}{64} \times \frac{3}{64} \times (\frac{61}{64})^{L-2} = \frac{3}{3721} \times (\frac{61}{64})^{L} = ka^{L}$$

Which is exponential in L.

The probability of an ORF appearing in a genome as a random sequence of codons is very small. Therefore, genes can be found by assuming ORFs with a length above some threshold value are there by evolutionary means; i.e. they are genes. To search for genes, a suitable method might be to look for a start codon ATG and create a sequence until a stop codon is reached as long as the sequence is longer than some threshold value.

The method presented in the given paper uses *de novo* gene prediction, which uses genomic DNA (that contains introns) to predict genes rather than also using cDNA (an expression-based prediction). Expression based prediction is limited by genomes where there is a lack of cDNA sequences to align with. Previous de novo prediction methods produced large amounts of false positive hits, but the TWINSCAN predictor uses a closely related genomic sequence to align with the given sequence to lower the false positive rate. Another major problem was that chromosomes needed to be split before processing, which resulted in genes that spanned multiple chromosomes being unrecognised. The gene prediction models discussed use hidden Markov models, where the states represent the function that a specific nucleotide performs a specific function (for instance, the third base of a codon).

2 Question 2

To identify CpG island regions in a genome, first construct a hidden Markov model using both T^{+} and T^{-} as subchains; it may be necessary to rename the states for clarity as to whether they are part of T^{+} or T^{-}. Transitions between the two sets of states can then be added using experimental data that determines the likelihoods for transitions

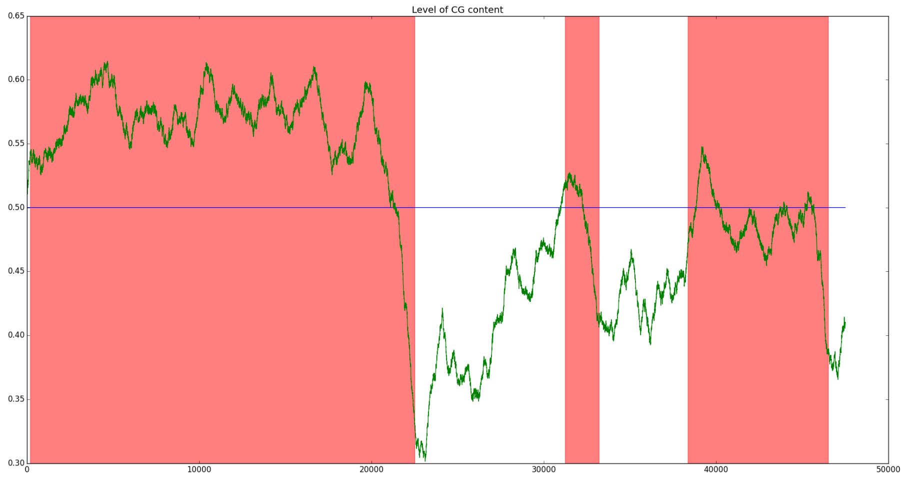

Figure 1: Moving average of last 1000 nucleotides measuring proportion of C and G nucleotides using a naive method (green line) and a HMM (red shaded regions indicate high CG content). Blue line is naive classifier.

between CpG island regions and non CpG island regions. Using the Viterbi algorithm on the given sequence and the newly constructed HMM, the most likely state path can be determined.

A quick way to find high CG content regions is to consider a moving average of the GC-ratio [?], calculated as:

$$\frac{G+C}{A+T+G+C}$$

A moving average window of the last 1000 nucleotides was used (Figure 1) to avoid both noise (Figure 2) and a curve which hides region information (Figure 2b). This region size is also a common size of CpG islands in the Hepatitis B virus genome [1], so provides a reasonable starting estimate for the size of high CG content regions. The horizontal line, y=0.5, is a naive classifier of high and low CG content regions. This is naive under the assumption that purines and pyrimidines would appear in equal proportions in a random RNA sequence (i.e. if there were no high or low regions of CG content, p(A) = p(G) = p(U/T) = p(C) = 0.25).

3 Question 3

The table is shown above. The most likely state sequence is H-H-L-L-H-H-H. All values are given to 3 d.p.

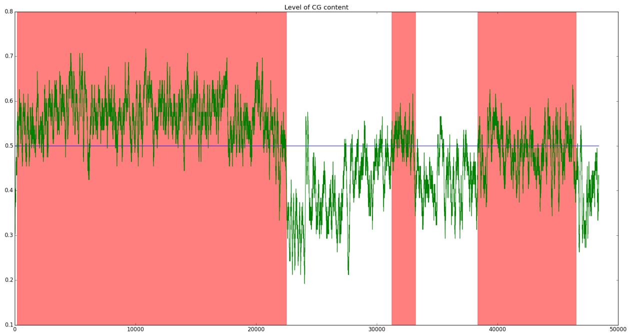

Figure 2: Using a window size of 100 nucleotides provides a noisy result.

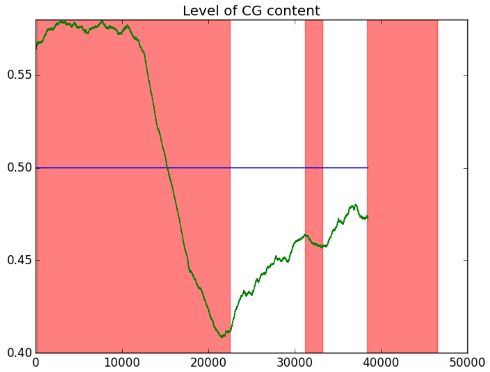

Figure 3: Using a window size of 10000 nucleotides is not granular enough to accurately provide region information.

Table 1: Delta values for the given sequence.

$\delta(t)$	Log Likelihood H	Log Likelihood L	Pointers
1	-2.889	-3.322	[]
2	-5.778	-6.211	[H]
3	- 9.094	-8.564	[H, H]
4	-11.836	-11.163	[H, H, L]
5	-14.101	-14.277	[H, H, L, L]
6	-16.990	-17.423	[H, H, L, L, H]
7	-19.776	-20.242	[H, H, L, L, H, H]

Using a HMM method, the shaded area in Figure 1 represents areas of 'high' CG content. In the given HMM, with transition probabilities p(AT-¿CG) = 0.0002 and p(CG-¿AT) = 0.0003 and initial state probabilities being equal, the red regions align with areas classified as high CG content using the naive classifier and the moving window approach. The main discrepancy is near the starts and ends of the high CG regions (e.g. around x=21750), where the naive method reports a low CG content but is still classified as a high CG content region by the HMM or vice versa.
A likely explanation is that the HMM 'lags' behind the true state change. As the differences in nucleotide probability are small between the two states and the transition probabilties a_{HL} and a_{LH} are also small, a long chain of nucleotides corresponding to the opposite state (A or T nucleotides for state L, G or C nucleotides for state H) are needed before the Viterbi algorithm calculates the state change.

b) To design a HMM to model the evolution of amino acid sequences, the best starting point would be a large data set of amino acid sequences that are linked via natural selection or otherwise mutated (an point accepted mutation), such as a PAM matrix [2]. In one of these matrices, the entry M(i, j) represnets the probability of amino acid j mutating into amino acid i over some defined time frame. Another option is a differently created substitution matrix such as a BLOSUM matrix [3]. However, the BLOSUM matrices are more suited to comparing evolutionary distant sequences, whereas Dayhoff's PAM matrices instead consider closely related species. The best result would be gained by comparing HMMs trained on both PAM and BLOSUM matrices to find the most effective fit for the situation. I am assuming that the modelling done would benefit more from the PAM matrices as the sequence inputs are closely related. Equally the specific PAM matrix (e.g. PAM10 is the matrix representing 10 mutations per 100 amino acids) to use would provide different results.

The transition probabilities are then the likelihood of substiting from one group of amino acids to a second group of amino acids:

$$p(S_t = polar \mid S_{t-1} = nonpolar) = \frac{\sum_{x \in nonpolar} \sum_{y \in polar} M(x,y)}{\sum_{x \in A} \sum_{y \in A} M(x,y)}$$

where A is the set of amino acids. A similar function can be constructed for other combinations of state transitions.

The emission probabilities for each state are then calculated by the proportion of each amino acid found within sequences (with potential tailoring to specific kinds of genome - for instance, human DNA likely has a different ratio of amino acid frequencies to that of E. coli), within the specific groups of polar and nonpolar amino acids:

$$p(observed = polar|S = nonpolar) = 0$$

and

$$\sum_{x \in polar} p(observed = x|S = polar) = 1$$

References

[1] Z. Shokrgozar, S. Tayebi, Z. Minucheher, and A. Mohamadkhani, "Hepatitis b virus genome asymmetry in hepatocellular carcinoma," *Middle East Journal of Digestive Diseases (MEJDD)*, vol. 4, no. 3, pp. 150–157, 2012.

[2] M. O. Dayhoff and R. M. Schwartz, "A model of evolutionary change in proteins," in *In Atlas of protein sequence and structure*, Citeseer, 1978.

[3] S. Henikoff and J. G. Henikoff, "Amino acid substitution matrices from protein blocks," *Proceedings of the National Academy of Sciences*, vol. 89, no. 22, pp. 10915–10919, 1992.