Profile Hidden Markov Model

Minkoo Seo
15 Oct. 2004
Data and Knowledge Engineering Lab.
Yonsei University

Profile

- Profile
  - Defined as a consensus primary structure model consisting of position-specific residue scores and insertion or deletion penalties
  - Based either on multiple sequence alignments or 3D structures
- Problems with profile
  - They are complicated models with many free parameters.
  - A number of difficult problems
    - What are the best ways to set the position specific residue scores, to score gaps and insertions, and to combine structural and multiple sequence information?
    - Until recently, these questions have generally been addressed in ad hoc fashion. (Eddy, 1996)

HMM (hidden Markov model)

- Very general form of probabilistic model for sequences of symbols
- Type of questions we can use HMM to consider are
  - Does this sequence belong to a particular family?
  - Assuming the sequence does come from some family, what can we say about its internal structure?

Example: CpG islands

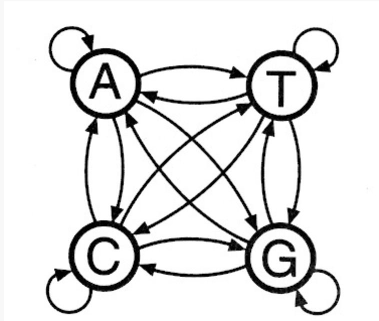
- In the human genome wherever the dinucleotide (A nucleotide molecule that consists of a combination of two nucleotide units) CG occurs, the C nucleotide is typically chemically modified by methylation.
- There is a relatively high chance of this methyl-C mutating into T.
- Consequently, in general CpG dinucleotides are rarer in genome than would be expected from the independent probabilities of C and G.

Example: CpG islands (cont)

- For biologically important reasons the <u>methylation process is suppressed in short stretches of the genome</u>, such as around the promoters or 'start' regions of many genes.
- In these regions, we see more CpG.
- Such regions are called *CpG islands*.
- We will consider two questions:
  - Given a short stretch of genomic sequence, how would we decide if it comes from a CpG island or not? → Markov chains
  - Given a long piece of sequence, how would we find the CpG islands is in it, if there are any? → hidden Markov models

Markov Chains

- We like to show a Markov chain graphically as a collections or 'states.'
- A Markov chain for DNA can be drawn like this



Markov Chains (cont)

- Transition probabilities

$$a_{st} = P(x_i = t \mid x_{i-1} = s)$$

- The probability of sequence can be obtained as

$$P(x) = P(x_L, x_{L-1}, \ldots x_1)$$
$$= P(x_L \mid x_{L-1}, \ldots x_1) P(x_{L-1} \mid x_{L-2}, \ldots, x_1) \ldots P(x_1)$$

by applying $P(X,Y) = P(X \mid Y)P(Y)$

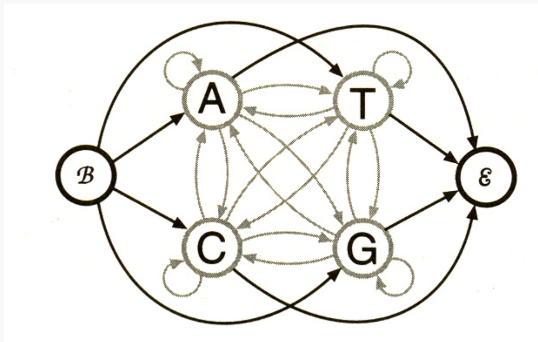
- In Markov chains, x_i depends only on the value of the preceding symbol x_{i-1}.

$$P(x) = P(x_L \mid x_{L-1}) P(x_{L-1} \mid x_{L-2}) \ldots P(x_2 \mid x_1) P(x_1)$$
$$= P(x_1) \prod_{i=2}^{L} a_{x_{i-1} x_i}$$

Markov Chains (cont)

- To avoid the inhomogeneity of transition probability, we add start and end states.

$$P(x_1 = s) = a_{\beta s}$$
$$P(\varepsilon \mid x_L = t) = a_{t\varepsilon}$$



Using Markov chains for discrimination

- A primary use for P(x) is to calculate the values for a likelihood ratio test.
- From a set of human DNA sequences, we extracted a total of 48 putative CpG islands and derived two Markov chain models, one for the regions labeled as CpG island (the '+' model) and the other from the remainder of the sequence (the '-' model).
- The transition probabilities for each model were set using the equation

$$a_{st}^{+} = \frac{c_{st}^{+}}{\sum_{t'} c_{st'}^{+}}$$

Number of times of any letter followed s.

and its analogue for '-' model, where c_{st} means is the number of times letter t followed s in the labeled regions.

Using Markov chains for discrimination (cont)

- Resulting tables are:

+	A	C	G	T
A	0.180	0.274	0.426	0.120
C	0.171	0.368	0.274	0.188
G	0.161	0.339	0.375	0.125
T	0.079	0.355	0.384	0.182

−	A	C	G	T
A	0.300	0.205	0.285	0.210
C	0.322	0.298	0.078	0.302
G	0.248	0.246	0.298	0.208
T	0.177	0.239	0.292	0.292

Using Markov chains for discrimination (cont)

- To use these models for discrimination, we calculate the log-odds ratio.

$$S(x) = \log \frac{P(x \mid \text{model+})}{P(x \mid \text{model-})} = \sum_{i=1}^{L} \log \frac{a_{x_{i-1}x_i}^{+}}{a_{x_{i-1}x_i}^{-}}$$

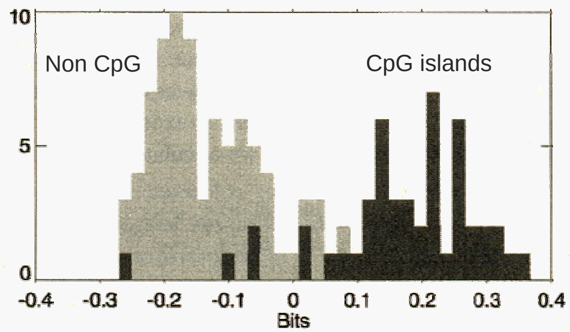
$$= \sum_{i=1}^{L} \beta x_{i-1} x_i$$

- Then, table for β is given as below in bits:

β	A	C	G	T
A	−0.740	0.419	0.580	−0.803
C	−0.913	0.302	1.812	−0.685
G	−0.624	0.461	0.331	−0.730
T	−1.169	0.573	0.393	−0.679

Using Markov chains for discrimination (cont)

- Following figure shows the distribution of scores, S(x), normalized by dividing by their length, i.e., as an average number of bits per molecule.
- If we had not normalized by length, the distribution would have been much more spread out.

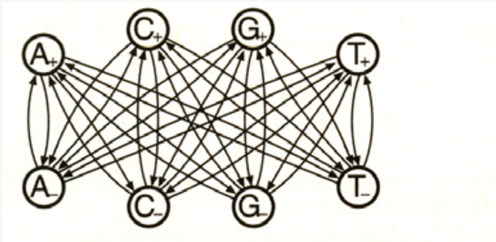


Hidden Markov Models

- How do we find CpG islands in a long sequence?
- The Markov chain we just built could be used for this purpose using a window of, say, 100 nucleotides.
- However, this is somewhat unsatisfactory is we believe that in fact CpG islands have sharp boundaries, and are of variable length.
- Why use a window size of 100?
- A more satisfactory approach is to build a single model for the entire sequence that incorporates both Markov chains.

Hidden Markov Models (cont)

- To simulate in one model the 'islands' and in a 'sea' of non-islands genomic sequence, we want to have both the Markov chains of previous slides in the same model with a small probability of switching from one chain to the other at each transition point.
- This introduce the complication that we now have two states corresponding to each nucleotide symbol.
- We resolve this by *relabelling the states*.



Hidden Markov Models (cont)

- The relabelling is the critical step.
- The essential difference between a Markov chain and a hidden Markov model is that for a hidden Markov model there is not a one-to-one correspondence between the states and the symbols.
- It is no longer possible to what state the model was in when x_i was generated just by looking x_i.

Formal definition of an HMM

- We now need to distinguish the sequence of states from the sequence of symbols.
- We now call the state sequence the path, π. *i*th state in the path is called π_i.
- The chain is characterized by parameters

$$a_{kl} = P(\pi_i = l \mid \pi_{i-1} = k)$$

cf. In Markov chain,

$$a_{st} = P(x_i = t \mid x_{i-1} = s)$$

where x_i is a character at *i*th position of x.

Formal definition of an HMM (cont)

- Because we have decoupled the symbols b from the state k, we must introduce a new set of parameters for the model, $e_k(b)$.
- In general, a state can produce a symbol from a distribution over all possible symbols. We therefore define

  $$e_k(b) = P(x_i = b \mid \pi_i = k)$$

  the probability that symbol b is seen when in state k.
- These are known as the *emission* probabilities.
- Joint probability of an observed sequence x and a state sequence π.

  $$P(x, \pi) = a_{0\pi_1} \prod_{i=1}^{L} e_{\pi_i}(x_i) a_{\pi_i \pi_{i+1}}$$

Formal definition of an HMM (cont)

- For example, the probability of sequence CGCG being emitted by the state sequence (C_+, G_-, C_-, G_+) in our model is

  $$(a_{0,C_+} \times 1) \times (a_{C_+,G_-} \times 1) \times (a_{G_-,C_-} \times 1) \times (a_{C_-,G_+} \times 1) \times a_{C_+,0}$$

- However, in general, $P(x,\pi)$ is not useful because we do not know the path.
- Hence, it is important to find *the path*.
- If we know the path, we can compute $P(x|M)$; a score of x in the model M.

The Viterbi algorithm

- This is a dynamic programming algorithm to find a best path.
- If we are to choose just one path for our prediction, perhaps the one with the highest probability should be chosen,

  $$\pi^* = \underset{\pi}{\operatorname{argmax}}\, P(x, \pi)$$

- Suppose that the probability $v_k(x_i)$ of the most probable path ending in state k with observation x_i is known for all states k. Then,

  $$v_l(x_{i+1}) = e_l(x_{i+1}) \max_k (v_k(x_i) a_{kl}) \qquad \pi_L^* = \underset{k}{\operatorname{argmax}}(\mathrm{v_k(L)a_{k0}})$$

- When we apply this algorithm to a longer sequence the derived optimal path π* will switch between the '+' and the '-' components of the model, and thereby give the precise boundaries of the predicted CpG island regions.

The forward algorithm

- We want to compute P(x) of a hidden Markov model.
  cf. P(x) of Markov chain

  $$P(x) = P(x_L \mid x_{L-1}) P(x_{L-1} \mid x_{L-2}) \ldots P(x_2 \mid x_1) P(x_1)$$
  $$= P(x_1) \prod_{i=2}^{L} a_{xi-1xi}$$

- We must add the probabilities for all possible paths

  $$P(x) = \sum_{\pi} P(x, \pi)$$

- This can be computed using the approach similar to the Viterbi algorithm.
- Surprisingly, in many cases

  $$P(x) \approx P(x \mid \pi^*)$$

Profile HMM

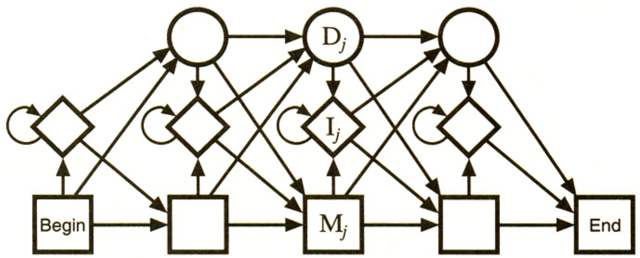


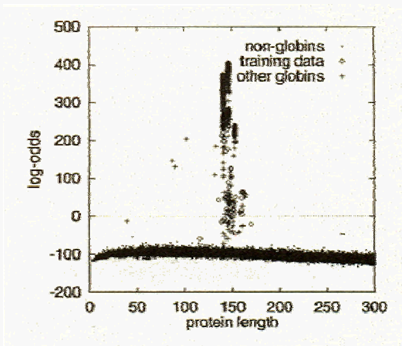
Figure 5.2 *The transition structure of a profile HMM. We use diamonds to indicate the insert states and and circles for the delete states.*

Profile HMM (cont)

- If we assume that we are looking for <u>global matches</u>, have a choice of ways to score a match to a hidden Markov model.
  - The Viterbi algorithm: $P(x,\pi^*|M)$
  - The forward algorith: $P(x|M)$

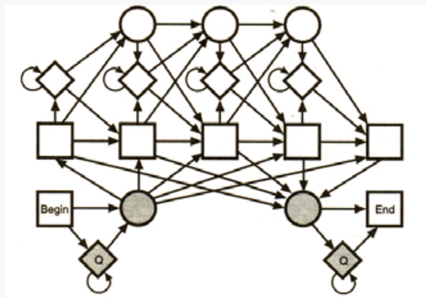
Profile HMM (cont)

- Example: Modelling and searching for globins
  - From 300 randomly picked globin sequences a profile HMM was built.
  - With this model a database of about 60,000 proteins was searched using the forward algorithm.



Profile HMM variants for non-global alignments

- We incorporates the original profile HMM together with one or more copies of a simple self-looping model.
- We call them *flanking model* states.
- The looping probability on the flanking states should be close to 1, since they must account for long stretches of sequence.



HMMER

- HMMER is an implementation of profile HMM methods for sensitive database searches using multiple sequence alignments as queries.
- Basically, you give HMMER a multiple sequence alignment as input; it builds a statistical model called a "hidden Markov model" which you can then use as a query into a sequence database to find (and/or align) additional homologues of the sequence family.
- hmms (“HMM Search”) looks for global alignments of the entire HMM to the entire query sequence.
- hmmsw (“HMM Smith/Waterman”) is an HMM version of the stand Smith/Waterman algorithm, allowing for local alignments that match any fragments of the HMM to any fragment of the query sequence.

Cost of HMM search algorithms

- With the exception of PSI-BLAST, profile HMM search algorithms are computationally demanding.
- Fast hardware implementations of Gribskov profile searches (Gribskov et al., 1987) are available.
- HMM approaches are also readily parallelized.
  - Intel Corporation has made a white paper available on using MMX assembly instructions to parallelize the Viterbi algorithm and get about a 2-fold speed increase on Intel hardware.

References

- Sean R Eddy, “Hidden Markov models,” Current Opinion in Structural Biology, 6:361-365, 1996.
- Sean R Eddy, “Profile hidden Markov models,” Bioinformatcis, 14(9):755-763, 1998.
- Anders Krogh, “An introduction to hidden Markov models for biological sequences,” In computational Methods in Molecular Biology, edited by S. L. Salzberg, D. B. Searls and S. Kasif, pp. 45-63, Elsevier, 1998.
- R. Durbin, S. Eddy, A. Krogh and G. Mitchison, BIOLOGICAL SEQUENCE ANALYSIS, Cambridge University Press, 1998.
- HMMER: profile HMMs for protein sequence analysis. http://hmmer.wustl.edu/
- Erik L. L. Sonnhammer et al, “Pfam: multiple sequence alignments and HMM-profiles of protein domains,” Nucleic Acids Research, 26(1):320-322, 1998.