

More on HMMs and Multiple Sequence Alignment

BMI/CS 776

www.biostat.wisc.edu/~craven/776.html

Mark Craven

craven@biostat.wisc.edu

March 2002

Announcements

- readings for the week after Spring break
 - Brown & Botstein, *Nature Genetics Supplement*
 - Eisen et al., *Proc. National Academy of Sciences*
 - and more

Multiple Sequence Alignment: Task Definition

- Given
 - a set of more than 2 sequences
 - a method for scoring an alignment
- Do:
 - determine the correspondences between the sequences such that the similarity score is maximized

Motivation

- characterizing a set of sequences (e.g. some class of DNA signals)
- characterizing a protein family
 - what is conserved
 - what varies
- building *profiles* for searching

Model Selection for Profile HMMs

- we have assumed we are given a model of a specified length; how do we determine this length?
- heuristic approach
 - choose an initial length; learn parameters
 - if more than $x\text{-del}\%$ of Viterbi paths go through delete state at position k , remove that position from model
 - if more than $x\text{-ins}\%$ go through insertions at position k , add new positions to the model
 - iterate

Classifying Sequences: Three Approaches

- choose threshold on $\Pr(x)$ that allows good discrimination between “positive” cases and “negative” cases
 - depends on length of x
- ✓ construct a “null” model; run query sequence x through both to see which results in greater $\Pr(x)$
- ✓ construct a set of models for disjoint families; run query sequence x through all models to see which results in highest $\Pr(x)$

Choosing a Threshold

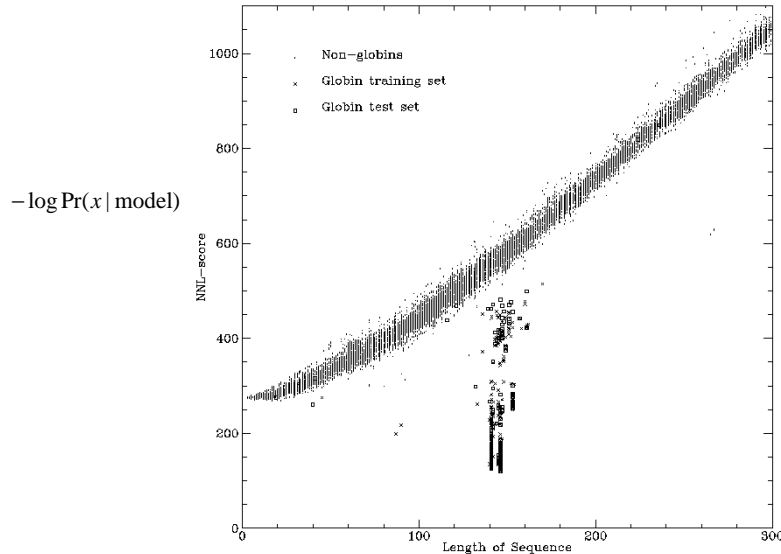
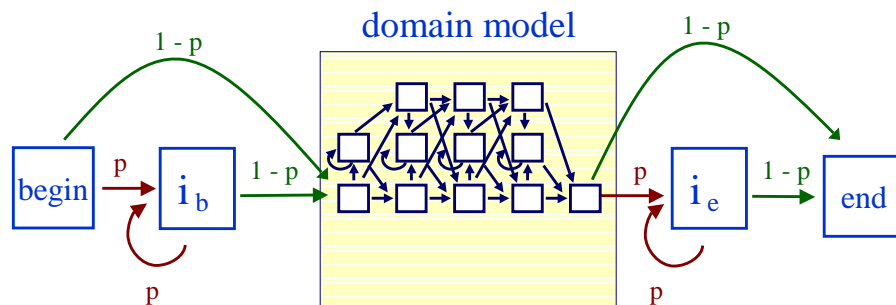


Figure from Krogh et al., Journal of Molecular Biology 235, 1994

Modeling Protein Domains with an HMM

- there are lots of ways we can modify the basic profile HMM architecture for particular modeling tasks – one such case is modeling protein *domains*



Other Methods: Scoring a Multiple Alignment

- key issue: how do we assess the quality of a multiple sequence alignment?
- usually, the assumption is made that the individual *columns* of an alignment are independent
- we'll discuss two methods
 - sum of pairs (SP)
 - minimum entropy

Scoring an Alignment: Sum of Pairs

- compute the sum of the pairwise scores

$$\text{Score}(m_i) = \sum_{k < l} s(m_i^k, m_i^l)$$

m_i^k = character of the k th sequence in the i th column

S = substitution matrix

Scoring an Alignment: Minimum Entropy

- basic idea: try to minimize the *entropy* of each column
- another way of thinking about it: columns that can be communicated using few bits are good
- information theory tells us that an optimal code uses $-\log_2 p$ bits to encode a message of probability p

Scoring an Alignment: Minimum Entropy

- the messages in this case are the characters in a given column
- the entropy of a column is given by:

$$\text{Score}(m_i) = -\sum_a c_{ia} \log_2 p_{ia}$$

m_i = the i th column of an alignment m

c_{ia} = count of character a in column i

p_{ia} = probability of character a in column i

Dynamic Programming Approach

- can find optimal alignments using dynamic programming
- generalization of methods for pairwise alignment
 - consider n -dimension matrix for n sequences (instead of 2-dimensional matrix)
 - each matrix element represents alignment score for n subsequences (instead of 2 subsequences)
- given n sequences of length L
 - space complexity is

$$O(L^n)$$

Dynamic Programming Approach

- given n sequences of length L
 - time complexity is

$$O(n^2 2^n L^n) \quad \text{if we use SP}$$

$$O(n 2^n L^n) \quad \text{if column scores can be computed in } O(n)$$

Heuristic Alignment Methods

- since complexity of DP approach is exponential in the number of sequences, heuristic methods are usually used
- *progressive alignment*: construct a succession of pairwise alignments
 - CLUSTALW
 - star approach
 - etc.
- iterative refinement
 - given a multiple alignment (say from a progressive method)
 - remove a sequence, realign it to profile of other sequences
 - repeat until convergence

Star Alignment Approach

- given: n sequences to be aligned
 x_1, \dots, x_n
 - pick one sequence x_c as the “center”
 - for each $x_i \neq x_c$ determine an optimal alignment between x_i and x_c
 - aggregate pairwise alignments
- return: multiple alignment resulting from aggregate

Star Alignments: Picking the Center

- try each sequence as the center, return the best multiple alignment
- compute all pairwise alignments and select the string that maximizes:

$$\sum_{i \neq c} \text{sim}(x_i, x_j)$$

Star Alignments: Aggregating Pairwise Alignments

- “once a gap, always a gap”
- shift entire columns when incorporating gaps

Star Alignment Example

	present pair	alignment
3.	<p>ATCTTC-TT ATTGCCATT</p>	<p>ATTGCCATT-- ATGGCCATT-- ATC-CAATTTT ATCTTC-TT--</p>
4.	<p>ATTGCCGATT ATTGCC-ATT</p>	<p>ATTGCC-ATT-- ATGGCC-ATT-- ATC-CA-ATTTT ATCTTC--TT-- ATTGCCGATT--</p>

Methods for Multiple Sequence Alignment

method	alignment types	search
multi-dimensional dynamic programming	global/local	dynamic programming
Star	global	greedy via pairwise alignments
CLUSTALW (tree)	global	greedy via pairwise alignment
profile HMMs	global/local	Baum-Welch (EM) to learn model, Viterbi to recover alignments
EM/MEME	local	EM

Probabilistic vs. Other Multiple Alignment Methods

- conventional methods use uniform substitution scores & gap penalties for all regions of sequences
- an HMM can score things differently in different regions (e.g. highly conserved vs. other regions)