Sequence alignment

Multiple alignment scoring and dynamic programming

Outline

- The multiple sequence alignment algorithm
- Scoring Multiple Sequence Alignments
- Dynamic programming

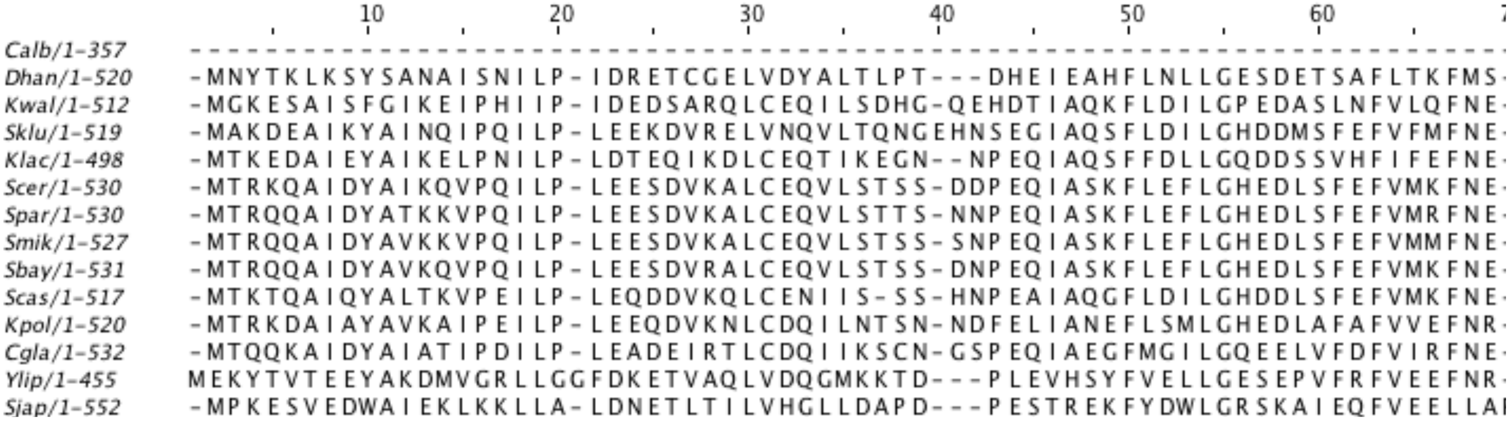
What is multiple sequence alignment?

Given: three or more related biological sequences

Do: identify the subsets of positions across sequences that are truly related

In other words: find a simultaneous alignment of all input sequences such that the implied pairwise alignments identify the truly related positions between each pair of sequences

An example multiple sequence alignment



```
                      10        20        30        40        50        60        7
Calb/1-357    ----------------------------------------------------------------------
Dhan/1-520    -MNYTKLKSYSANAISNILP-IDRETCGELVDYALTLPT---DHEIEAHFLNLLGESDETSAFLTKFMS-
Kwal/1-512    -MGKESAISFGIKEIPHIIP-IDEDSARQLCEQILSDHG-QEHDTIAQKFLDILGPEDASLNFVLQFNE-
Sklu/1-519    -MAKDEAIKYAINQIPQILP-LEEKDVRELVNQVLTQNGEHNSEGIAQSFLDILGHDDMSFEFVFMFNE-
Klac/1-498    -MTKEDAIEYAIKELPNILP-LDTEQIKDLCEQTIKEGN--NPEQIAQSFFDLLGQDDSSVHFIFEFNE-
Scer/1-530    -MTRKQAIDYAIKQVPQILP-LEESDVKALCEQVLSTSS-DDPEQIASKFLEFLGHEDLSFEFVMKFNE-
Spar/1-530    -MTRQQAIDYATKKVPQILP-LEESDVKALCEQVLSTTS-NNPEQIASKFLEFLGHEDLSFEFVMRFNE-
Smik/1-527    -MTRQQAIDYAVKKVPQILP-LEESDVKALCEQVLSTSS-SNPEQIASKFLEFLGHEDLSFEFVMMFNE-
Sbay/1-531    -MTRQQAIDYAVKQVPQILP-LEESDVRALCEQVLSTSS-DNPEQIASKFLEFLGHEDLSFEFVMKFNE-
Scas/1-517    -MTKTQAIQYALTKVPEILP-LEQDDVKQLCENIIS-SS-HNPEAIAQGFLDILGHDDLSFEFVMKFNE-
Kpol/1-520    -MTRKDAIAYAVKAIPEILP-LEEQDVKNLCDQILNTSN-NDFELIANEFLSMLGHEDLAFAFVVEFNR-
Cgla/1-532    -MTQQKAIDYAIATIPDILP-LEADEIRTLCDQIIKSCN-GSPEQIAEGFMGILGQEELVFDFVIRFNE-
Ylip/1-455    MEKYTVTEEYAKDMVGRLLGGFDKETVAQLVDQGMKKTD---PLEVHSYFVELLGESEPVFRFVEEFNR-
Sjap/1-552    -MPKESVEDWAIEKLKKLLA-LDNETLTILVHGLLDAPD---PESTREKFYDWLGRSKAIEQFVEELLAI
```

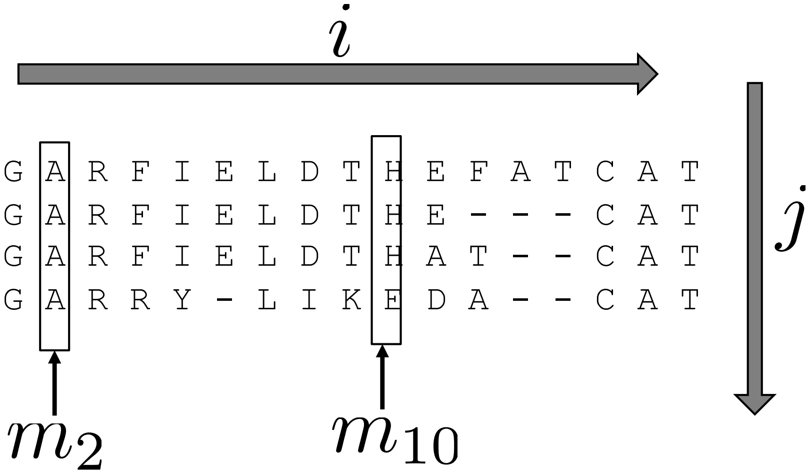
Why multiple sequence alignment?

- Build phylogenetic trees (next module)
  - Determine evolutionary relationships between sequences
- A multiple sequence alignment can represent a family of proteins with similar function
  - Compare new sequence to a “family” of known proteins
  - For example the BLOCKS database used for BLOSUM contains several ungapped alignments for known protein families
- Discover common signatures or protein domains among a group of proteins
- Identify genetic variation among individuals of a population

Some notation for multiple sequence alignments

- Let m denote a Multiple Sequence Alignment
- m_i is the i^{th} column of the alignment m
- m_i^j is the i^{th} column and j^{th} row
- c_i^a count of residue a in column i

Example using notation



```
G A R F I E L D T H E F A T C A T
G A R F I E L D T H E - - - C A T
G A R F I E L D T H A T - - C A T
G A R R Y - L I K E D A - - C A T
```

$$m_3^1 = R$$

$$c_3^A = 0$$
$$c_2^A = 4$$
$$c_{10}^H = 3$$

Scoring a Multiple Sequence Alignment (MSA)

- Key issue: how do we score a multiple sequence alignment?
- Usually, we assume that *columns* of an alignment are independent

$$Score(m) = G(m) + \sum_i S(m_i)$$

gap function

score of *i*th column

- For now, we will simplify the score by assuming a linear gap penalty

$$Score(m) = \sum_i S(m_i)$$

Gap penalty (G)

- We will use a simple linear gap penalty function
  - Penalty for a space: s
- Let S(a,b) denote the cost of substituting a by b.
- Linear gap penalty can be incorporated into the substitution matrix
  - S(a,-) = S(-, a) = s
  - S(-,-)=0

Two common ways of scoring a multiple alignment

- Entropy based scores

- Sum of pairs

Entropy of a distribution

- A measure of uncertainty of an outcome
- For a discrete distribution *P(X)*, where *X* takes k values x_1, .. x_k it is defined as

$$H(X) = -\sum_{i=1}^{k} P(x_i)\mathrm{log}P(x_i)$$

- Entropy is greatest when we are most uncertain, that is, for a uniform distribution
- Entropy is least when we are most certain, e.g. deterministic event

Entropy in extreme cases

Score of a column: Entropy based

- Score of the i^{th} column of alignment m is

$$S(m_i) = -\sum_a c_i^a \log(p_{ia})$$

$$p_{ia} : \text{Probability of character } a \text{ in column } i$$

$$c_i^a : \text{Number of occurrences of } a \text{ in column } i$$

- This has an entropy-based interpretation
  - Let X_i be a random variable representing a character in column i
  - Consider each entry of column i to be observations of X_i across multiple independent experiments
  - We estimate $P(X_i = a)$ by $p_{ia} = \dfrac{c_i^a}{n}$
  - Column score is proportional to the entropy of X_i

Scoring an alignment: Entropy based score

- High entropy: More uniform distribution/more variability of characters
- Low entropy: Less uniform distribution/less variability of characters

$$S(m_i) = -\sum_a c_i^a \log(p_{ia})$$

Scoring of a column: Sum of Pairs

- Compute the sum of the pairwise scores

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

Iterate over all pairs of rows in the column

$s(m_i^k, m_i^l)$ Substitution score from a substitution/match matrix such as BLOSUM or PAM

Dynamic Programming (DP) for global multiple sequence alignment

- Assume columns are independent
  - Score of alignment is sum of column scores
- Generalization of methods for pairwise alignment
  - consider k-dimensional matrix for k sequences (instead of 2-dimensional matrix)
  - each matrix element represents alignment score for k prefixes (instead of 2 prefixes)

Notation for DP

- Assume we have k sequences $x^1, \cdots, x^k$
- i_1 denotes the length of the prefix for sequence 1
- i_2 denotes the length of the prefix for sequence 2
- $\ldots$
- i_k denotes the length of the prefix for sequence k
- $x^k_{i_k}$ denotes the character at i_k position of sequence x^k
- F: k-dimensional matrix where

$$F(i_1, i_2, \cdots, i_k)$$

denotes the score of the best alignment of the i_1, i_2.. i_k prefixes of the sequences

Recall the DP for the pairwise alignment

$$F(i_1, i_2) = \max \begin{cases} F(i_1 - 1, i_2 - 1) + S(x^1_{i_1}, x^2_{i_2}) \\ F(i_1, i_2 - 1) + S(-, x^2_{i_2}) \\ F(i_1 - 1, i_2) + S(x^1_{i_1}, -) \end{cases}$$

DP for Multiple sequence alignment

$$F(i_1,\cdots,i_k) = \max\begin{cases} F(i_1-1,\cdots,i_k-1)+S(x^1_{i_1},\cdots,x^k_{i_k}) \\ F(i_1,i_2-1,\cdots,i_k-1)+S(-,x^2_{i_2},\cdots,x^k_{i_k}) \\ F(i_1-1,i_2,\cdots,i_k-1)+S(x^1_{i_1},-,\cdots,x^k_{i_k}) \\ \vdots \\ F(i_1,i_2-1,\cdots,i_k)+S(-,x^2_{i_2},\cdots,-) \\ \vdots \end{cases}$$

max score of alignment for the *k* prefixes

How many items do we need to maximize over? 2^k *-1*

DP algorithm is too expensive

- For k sequences each of length *n*
  - Space complexity: $O(n^k)$
  - Time complexity: $O(n^k 2^k)$

Summary

- Multiple alignment task and applications
- Two scoring functions
  - Entropy-based
  - Sum of pairs
- Inefficient dynamic programming extension