Pairwise Alignment with HMMs

An HMM for denoting gap - matches, mismatches

The most probable path is the optimal alignment.

Viterbi algorithm:

Let the "Begin" state be "M".

- \( v^*(i,j) \) : probabilities
- \( V^*(i,j) \) : log of probabilities

Initialize: \( V^m(0,0) = 1 \), all other \( v^*(i,0), v^*(0,i) = 0 \)

Recurrence: \( i = 1 \ldots n, j = 1 \ldots m \)

\[
V^m(i,j) = P_{xiyj} \max \left\{ (1-2\delta-2\gamma) V^m(i-1,j-1), (1-\delta-3\gamma) V^x(i-1,j-1), (1-\gamma-3\delta) V^y(i-1,j-1) \right\}
\]

\[
v^x(i,j) = q_{xj} \max \left\{ \delta V^m(i-1,j), \varepsilon V^x(i-1,j) \right\}
\]

\[
v^y(i,j) = q_{yj} \max \left\{ \delta V^m(i,j-1), \varepsilon V^y(i,j-1) \right\}
\]

HMM with above Viterbi algorithm will give the exact same output with NW algorithm:

\[
v^E = 2 \max (V^m(n,m), V^x(n,m), V^y(n,m))
\]
If one wants to be able to generate sequences with variable length (affine gap weights) then one should have:

![Diagram](image)

This creates $x_i$'s and $y_j$'s independently. According to this model:

$$\Pr(x, y | R) = \eta^n (1 - \eta)^m \left[ \prod_{i=1}^{n} q_{x_i} \right] \eta^{m} \left[ \prod_{j=1}^{m} q_{y_j} \right]$$

One can then define:

$$s(a,b) = \log \frac{p_{ab}}{q_a q_b} + \log \frac{(1 - 2 \beta - 2)}{(1 - \eta)^2}$$

$$d = -\log \frac{s(1 - 3 - 2)}{(1 - \eta)(1 - 2 - 2)}$$

$$e = -\log \frac{e}{1 - 2}$$

to write down Viterbi's algorithm in standard DP formulation:

Initialize. $V^m(0,0) = 2 \log \eta$, $V^x(0,0) = V^y(0,0) = -\infty$

All $V^*(i,j-1) = V^*(i-1,j) = -\infty$

$$V^m(i,j) = s(x_i, y_j) + \max \left\{ V^m(i-1,j-1) + d, V^x(i-1,j-1) + e \right\}$$

$$V^x(i,j) = \max \left\{ V^m(i-1,j) - d, V^x(i-1,j) - e \right\}$$

$$V^y(i,j) = \max \left\{ V^m(i,j-1) - d, V^y(i,j-1) - e \right\}$$

$$V = \max (V^*(n,m))$$
Multiple Sequence Alignment via HMMs

Profile HMM. Define match states $M_1, \ldots, M_L$ which will be those states that would match a given position of any string to the profile. Also define insertion states $I_1, \ldots, I_L$ and deletion states $D_1, \ldots, D_L$.

Given string $x_1, \ldots, x_m$, one can match it against this "profile" HMM as follows:

* let $V_{M_j}(i)$ be the log probability of the "best path" of matching $x_1, \ldots, x_i$ to the HMM ending with $M_j$ - emitting $x_i$.
* Similarly $V_{I_j}(i)$ be the log probability of the "best path" of matching $x_1, \ldots, x_i$ to the HMM ending with $I_j$ - emitting $x_i$.
* Finally let $V_{D_j}(i)$ be the log probability of the "best path" ending with $D_j$ - no emission.
\[ v_{\text{begin}}(0) = 0 \quad \text{(these are log probabilities)} \]

Then:
\[ v_{M_j}(i) = \log e_{M_j}(x_i) + \max \left\{ v_{M_j}(i-1) + \log(a_{M_j-1,M_j}), v_{I_j}(i-1) + \log(a_{I_j-1,M_j}), v_{D_j}(i-1) + \log(a_{D_j-1,M_j}) \right\} \]
\[ v_{I_j}(i) = \log e_{I_j}(x_i) + \max \left\{ v_{M_j}(i-1) + \log(a_{M_j-1,I_j}), v_{I_j}(i-1) + \log(a_{I_j-1,I_j}), v_{D_j}(i-1) + \log(a_{D_j-1,I_j}) \right\} \]
\[ v_{D_j}(i) = \max \left\{ v_{M_j}(i) + \log(a_{M_j-1,D_j}), v_{I_j}(i) + \log(a_{I_j-1,D_j}), v_{D_j}(i-1) + \log(a_{D_j-1,D_j}) \right\} \]

Score \((X)\) = \max \left\{ v_{M_j}(m) + \log(a_{M_j-1,\text{end}}), v_{I_j}(m) + \log(a_{I_j-1,\text{end}}), v_{D_j}(m) + \log(a_{D_j-1,\text{end}}) \right\} \]

**Forward - Backward Probabilities for Profile HMM**

\[ f_{M_j}(i) = \log [\Pr(X_1, \ldots, x_i; \text{ending at } M_j)] \]
\[ f_{I_j}(i) = \log [\Pr(X_1, \ldots, x_i; \text{ending at } I_j)] \]
\[ f_{D_j}(i) = \log [\Pr(X_1, \ldots, x_i; \text{ending at } D_j)] \]

\[ b_{M_j}(i) = \log [\Pr(x_{i+1}, \ldots, x_m; \text{beginning at } M_j)] \]
\[ b_{I_j}(i) = \log [\Pr(x_{i+1}, \ldots, x_m; \text{beginning at } I_j)] \]
\[ b_{D_j}(i) = \log [\Pr(x_{i+1}, \ldots, x_m; \text{beginning at } D_j)] \]
Recursion for forward probabilities:
\[
\begin{align*}
  f_{m_j}(i) &= \log(e_{m_j}(x_i)) + \log \left[ a_{m_j-1,m_j} \cdot 2 f_{m_j-1}(i-1) \right. \\
  & \quad + a_{I_{j-1},m_j} \cdot 2 f_{I_{j-1}}(i-1) \] \\
  & \quad \left. + a_{D_{j-1},m_j} \cdot 2 f_{D_{j-1}}(i-1) \right] \\
  f_{I_{j}}(i) &= \log(e_{I_{j}}(x_i)) + \log \left[ a_{m_j-1,I_{j}} \cdot 2 f_{m_j-1}(i-1) \right. \\
  & \quad + a_{I_{j-1},I_{j}} \cdot 2 f_{I_{j-1}}(i-1) \] \\
  & \quad \left. + a_{D_{j-1},I_{j}} \cdot 2 f_{D_{j-1}}(i-1) \right] \\
  f_{D_{j}}(i) &= \log \left[ a_{m_j-1,D_{j}} \cdot 2 f_{m_j-1}(i-1) \right. \\
  & \quad + a_{I_{j-1},D_{j}} \cdot 2 f_{I_{j-1}}(i-1) \] \\
  & \quad \left. + a_{D_{j-1},D_{j}} \cdot 2 f_{D_{j-1}}(i-1) \right]
\end{align*}
\]

Backward probabilities are derived similarly.

Bauu - Welch procedure

\[ \hat{r}_{\text{estimates}} \] emission
\[
\begin{align*}
  \hat{E}_{m_j}(a) &= \frac{1}{\Pr(x_1 \ldots x_m)} \sum_{j \mid x_j = a} f_{m_j}(j) \cdot b_{m_j}(j) \\
  \hat{E}_{I_{j}}(a) &= \frac{1}{\Pr(x_1 \ldots x_m)} \sum_{j \mid x_j = a} f_{I_{j}}(j) \cdot b_{I_{j}}(j)
\end{align*}
\]
for $S_i = m_i, I_i$ or $D_i$:

$$A_{S_i, m_{i+1}} = \frac{1}{Pr(x_{i...n})} \sum_{v_j} f_{S_i}(j) a_{S_i, m_{i+1}} e_{m_{i+1}}(x_{j+1}) b_{m_{i+1}}(j)$$

$$A_{S_i, I_i} = \frac{1}{Pr(x_{i...n})} \sum_{v_j} f_{S_i}(j) a_{S_i, I_i} e_{I_i}(x_{j+1}) b_{I_i}(j+1)$$

$$A_{S_i, D_{i+1}} = \frac{1}{Pr(x_{i...n})} \sum_{v_j} f_{S_i}(j) a_{S_i, D_{i+1}} b_{D_{i+1}}(j)$$

Given a profile HMM $P$, it is possible to use it for multiply aligning sequences $X_1...X_n$:

1) Align each $X_i$ separately to $P$
2) Merge the alignments into one joint alignment by having a unique column for each insert.

If $P$ is not known:

1) Choose a length $L$ for the profile and initialize the emission and transition probabilities
2) Apply B-W algorithm to "learn" the correct parameters
3) Apply the above method for aligning the sequences to each other.
More Heuristics:

Gibbs Sampling for locating a common pattern/motif and finding a local multiple alignment

Input: sequences $S_1, \ldots, S_n$; integers $w$

Output: Find for each $S_i$ a substring of at most $w$ characters so that the similarity between $n$ substrings are maximized.

1) let $q_i, \ldots, q_n$ be starting positions of these substrings.
2) let $c_{ij}$ be the number of occurrences of symbol $j$ among the $i^{th}$ positions of $n$ substrings.
3) let $q_{ij}$ be the probability of symbol $j$ to occur at the $i^{th}$ position of the pattern.
4) let $p_j$ be the frequency of the symbol $j$ in all sequences of $S$.

We would like to maximize:

$$\text{Score} = \sum_{i=1}^{n} \sum_{j \in \mathcal{S}} c_{ij} \log \frac{q_{ij}}{p_j}$$

We follow the iterative procedure:

1) Randomly choose $q_1, \ldots, q_n$.
2) Randomly choose $1 \leq z \leq n$ and calculate $c_{ij}, q_{ij}$ and $p_j$ values for the strings $S \setminus S_z$.
3) Find best substring of $S_z$ according to the model and determine new $q_z$ by applying local alignment of $S_z$ against profile of current pattern.
4) Repeat steps 2 & 3 until improvement is small.