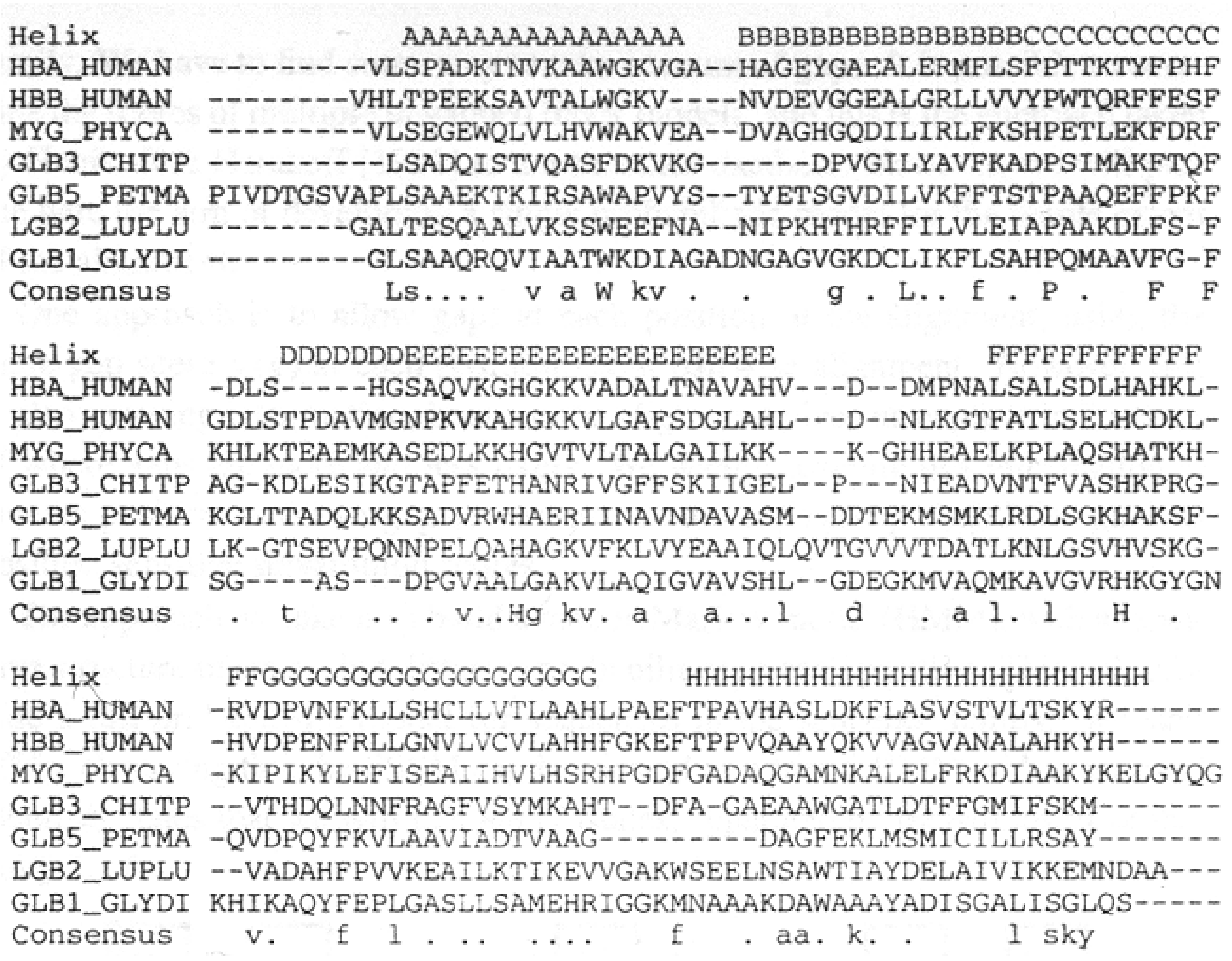
Searching for family members - (Durbin et al., Ch.5)

- Suppose we have a family of related sequences
  - interested in searching the db for additional members
- Lazy ideas:
  - choose a member
  - try all members
- In either case we are loosing information
  - better: combine information from all members
- The first step is to create a multiple alignment

Multiple alignment of seven globins



```
Helix                 AAAAAAAAAAAAAAAA   BBBBBBBBBBBBBBBBCCCCCCCCCCC
HBA_HUMAN   ---------VLSPADKTNVKAAWGKVGA--HAGEYGAEALERMFLSFPTTKTYFPHF
HBB_HUMAN   --------VHLTPEEKSAVTALWGKV----NVDEVGGEALGRLLVVYPWTQRFFESF
MYG_PHYCA   ---------VLSEGEWQLVLHVWAKVEA--DVAGHGQDILIRLFKSHPETLEKFDRF
GLB3_CHITP ----------LSADQISTVQASFDKVKG------DPVGILYAVFKADPSIMAKFTQF
GLB5_PETMA PIVDTGSVAPLSAAEKTKIRSAWAPVYS--TYETSGVDILVKFFTSTPAAQEFFPKF
LGB2_LUPLU --------GALTESQAALVKSSWEEFNA--NIPKHTHRFFILVLEIAPAAKDLFS-F
GLB1_GLYDI ---------GLSAAQRQVIAATWKDIAGADNGAGVGKDCLIKFLSAHPQMAAVFG-F
Consensus            Ls....  v a W kv .  .    g . L.. f . P .   F  F

Helix          DDDDDDEEEEEEEEEEEEEEEEEEEEEE            FFFFFFFFFFFF
HBA_HUMAN  -DLS-----HGSAQVKGHGKKVADALTNAVAHV---D--DMPNALSALSDLHAHKL-
HBB_HUMAN  GDLSTPDAVMGNPKVKAHGKKVLGAFSDGLAHL---D--NLKGTFATLSELHCDKL-
MYG_PHYCA  KHLKTEAEMKASEDLKKHGVTVLTALGAILKK----K-GHHEAELKPLAQSHATKH-
GLB3_CHITP AG-KDLESIKGTAPFETHANRIVGFFSKIIGEL--P---NIEADVNTFVASHKPRG-
GLB5_PETMA KGLTTADQLKKSADVRWHAERIINAVNDAVASM--DDTEKMSMKLRDLSGKHAKSF-
LGB2_LUPLU LK-GTSEVPQNNPELQAHAGKVFKLVYEAAIQLQVTGVVVTDATLKNLGSVHVSKG-
GLB1_GLYDI SG----AS---DPGVAALGAKVLAQIGVAVSHL--GDEGKMVAQMKAVGVRHKGYGN
Consensus   . t    .. . v..Hg kv. a   a...l   d   . a l. l   H  .

Helix       FFGGGGGGGGGGGGGGGGGGG     HHHHHHHHHHHHHHHHHHHHHHHHHHH
HBA_HUMAN  -RVDPVNFKLLSHCLLVTLAAHLPAEFTPAVHASLDKFLASVSTVLTSKYR------
HBB_HUMAN  -HVDPENFRLLGNVLVCVLAHHFGKEFTPPVQAAYQKVVAGVANALAHKYH------
MYG_PHYCA  -KIPIKYLEFISEAIIHVLHSRHPGDFGADAQGAMNKALELFRKDIAAKYKELGYQG
GLB3_CHITP --VTHDQLNNFRAGFVSYMKAHT--DFA-GAEAAWGATLDTFFGMIFSKM-------
GLB5_PETMA -QVDPQYFKVLAAVIADTVAAG---------DAGFEKLMSMICILLRSAY-------
LGB2_LUPLU --VADAHFPVVKEAILKTIKEVVGAKWSEELNSAWTIAYDELAIVIKKEMNDAA---
GLB1_GLYDI KHIKAQYFEPLGASLLSAMEHRIGGKMNAAAKDAWAAAYADISGALISGLQS-----
Consensus    v.   f  l . ..  ....    f   . aa. k. .     l sky
```

<gaps><learning>

Profile and Position Specific Scoring Matrix

- In this section we assume the alignment is given
    - by structure alignment or multiple sequence alignment
- Ignore insertions/deletions for now
- Each position in the alignment has its own “profile” of conservation
- How do we score a sequence aligned to the family?
- Use these conservation profiles to define PSSMs, or Position Specific Scoring Matrices

Gribskov et al.'s PSSMs (87)

- One approach is to average the contributions from the substitution matrix:

$$s_i(k) = \sum_j \alpha_{ij} S(k,j)$$

  - α_{ij} is the frequency of the jth AA at the ith position
  - $S(k,j)$ is the score of substituting AA k with j

- If the family contains just one sequence (pairwise alignment) the profile degenerates to one letter, x_i, and

$$s_i(k) = S(k, x_i)$$

  - which is exactly the scoring matrix we use for pairwise alignment

- A downside of this approach is that it fails to distinguish between a degenerate position 100 letters "deep" vs. 1 letter deep

HMM's derived PSSMs (Haussler et al. 93)

- An alternative approach is to think about the positions as states in an HMM each with its own emission profile: $p(\boldsymbol{x}) = \prod_i e_i(x_i)$
  - At this point there is nothing hidden about this HMM
- To test for family membership we can evaluate the log-odds ratio

$$S = \sum_i \log \frac{e_i(x_i)}{q(x_i)}$$

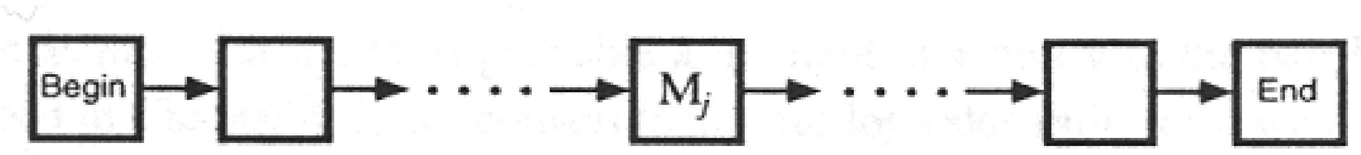
  - the PSSM $s_i(x) := \log \frac{e_i(x)}{q(x)}$ replaces the substitution matrix
- The emissions probabilities can be quite flexible
  - For example, in the case of a 1-sequence family we can set $e_i(x) := \frac{p(x,x_i)}{q(x_i)}$
    - ▷ where $p(x, y)$ is the joint probabilty from BLOSUM
  - and $s_i(x) = \log \frac{p(x,x_i)}{q(x)q(x_i)} = S(x, x_i)$ as for pairwise alignment

Mind the gap

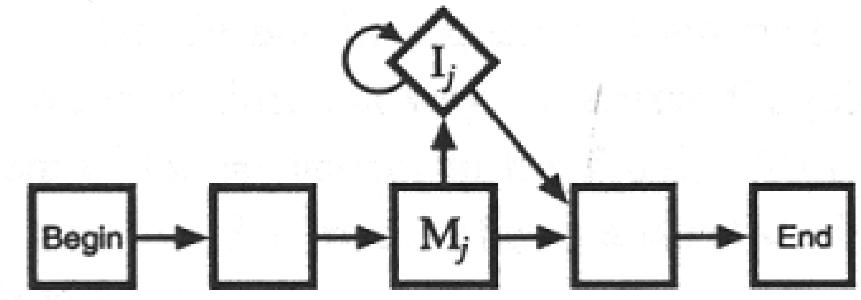
- How should we handle gaps?
- Gribskov et al. suggested a heuristic that decreased the cost of a gap (insertion or deletion) according to the length of the longest gap, in the multiple alignment, that spanned that column
    - this (again) ignores the popularity of the gap <globins>
- Alternatively, we can build a generative model that allows gaps

“Evolution” of profile HMMs

- Profiles without gaps; match states emit according to $e_M(x)$



- Allowing insertions; for insert states emissions $e_I(x) = q(x)$ typically



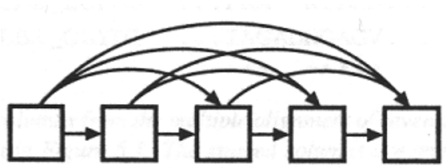
  - using llr the score contribution of a k letter insert is

$$\log a_{M_j I_j} + (k-1) \log a_{I_j I_j} + \log a_{I_j M_j}$$

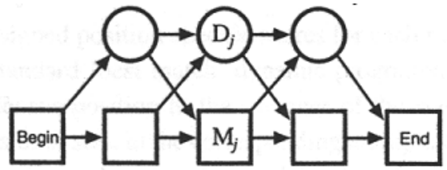
corresponding to an affine gap penalty (in pairwise alignment)

Evolution of profile HMMs - cont.

- Allowing for deletions

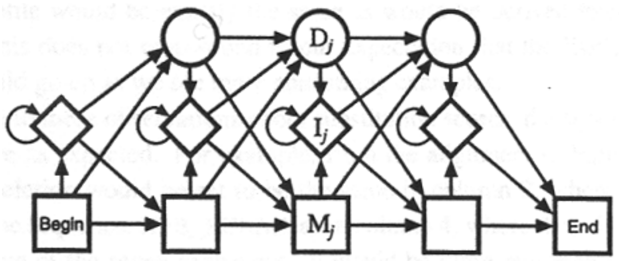


- Too many parameters: recall the silent states



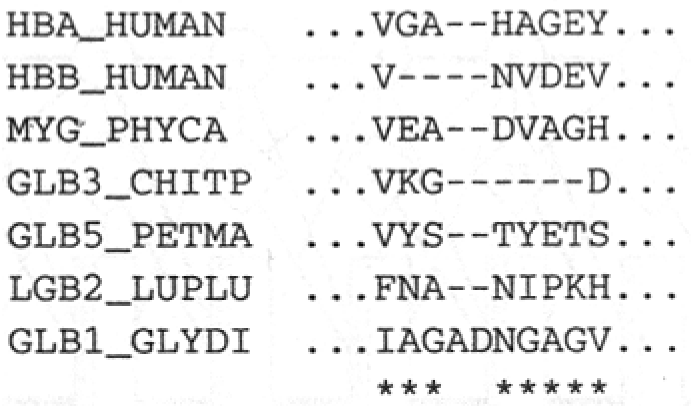
  - the cost of $D_i \rightarrow D_{i+1}$ can vary

- Profile HMMs (Haussler et al. 93):



Deriving profile HMMs from multiple alignment

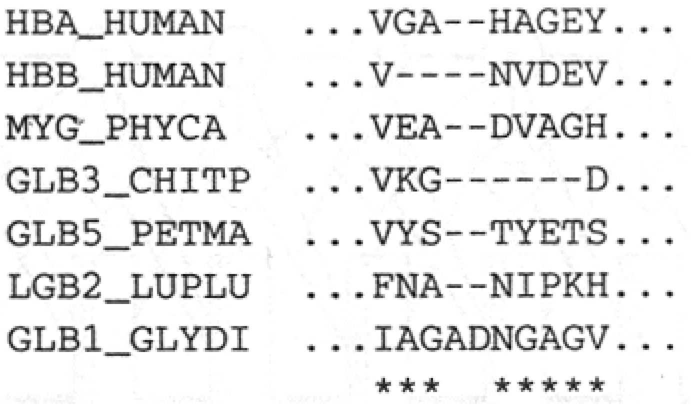
- The first problem in deriving the profile HMM is that of determining the length, or the number of gap states <globins>
- Heuristic: a column is a match state if it contains $< 50\%$ gaps
  - for example



```
HBA_HUMAN    ...VGA--HAGEY...
HBB_HUMAN    ...V----NVDEV...
MYG_PHYCA    ...VEA--DVAGH...
GLB3_CHITP   ...VKG------D...
GLB5_PETMA   ...VYS--TYETS...
LGB2_LUPLU   ...FNA--NIPKH...
GLB1_GLYDI   ...IAGADNGAGV...
                ***  *****
```

- With the topology of the HMM given the path generating every sequence in the family is determined
- We can use maximum-likelihood with pseudo-counts to estimate the parameters: a_{kl} and $e_k(x)$

Example of parameters estimation



```
HBA_HUMAN   ...VGA--HAGEY...
HBB_HUMAN   ...V----NVDEV...
MYG_PHYCA   ...VEA--DVAGH...
GLB3_CHITP  ...VKG------D...
GLB5_PETMA  ...VYS--TYETS...
LGB2_LUPLU  ...FNA--NIPKH...
GLB1_GLYDI  ...IAGADNGAGV...
               ***  *****
```

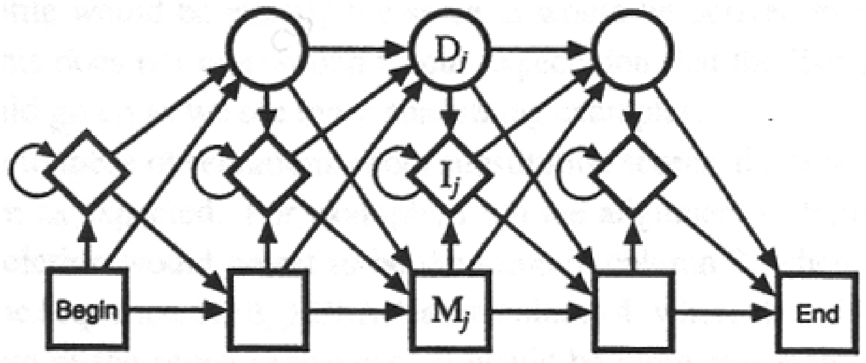
- Using Laplace's rule (add a pseudocount of 1 to each count) we have, for example, for the emission probabilities at M_1:

$$e_{M_1}(X) = \begin{cases} \frac{6}{27} & X = \mathtt{V} \\ \frac{2}{27} & X \in \{\mathtt{I},\mathtt{F}\} \\ \frac{1}{27} & X = \text{AA other than } \mathtt{V},\ \mathtt{I},\ \mathtt{F} \end{cases}$$

- Similarly, using the same pseudocounts, we estimate the transitions out of M_1 by:$a_{M_1M_2} = \frac{7}{10}$, $a_{M_1D_2} = \frac{2}{10}$, and $a_{M_1I_2} = \frac{1}{10}$

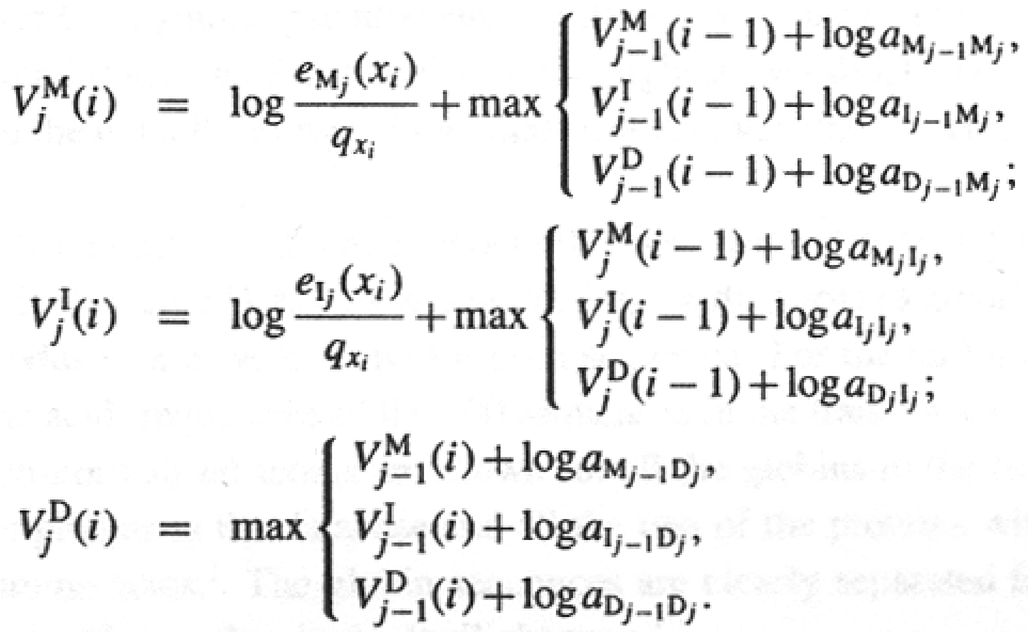
Searching with profile HMMs

- To determine whether or not a new sequence belongs to the family we need a similarity criterion
  - analogous to the similarity score Needleman-Wunsch optimizes
  - We can ask for the joint probability of the ML path and the data
  - or, for the probability of the data given the model
  - In either case for practical purposes log-odds ratio is prefferable
- Reminder: profile HMMs



Viterbi equations (from Durbin et al.)

- Let $V_j^s(i)$ be the log-odds ratio of the best path matching $\boldsymbol{x}_{1:i}$ to the model that ends at state s_j ($s \in \{M, D, I\}$). For $j \geq 1$:

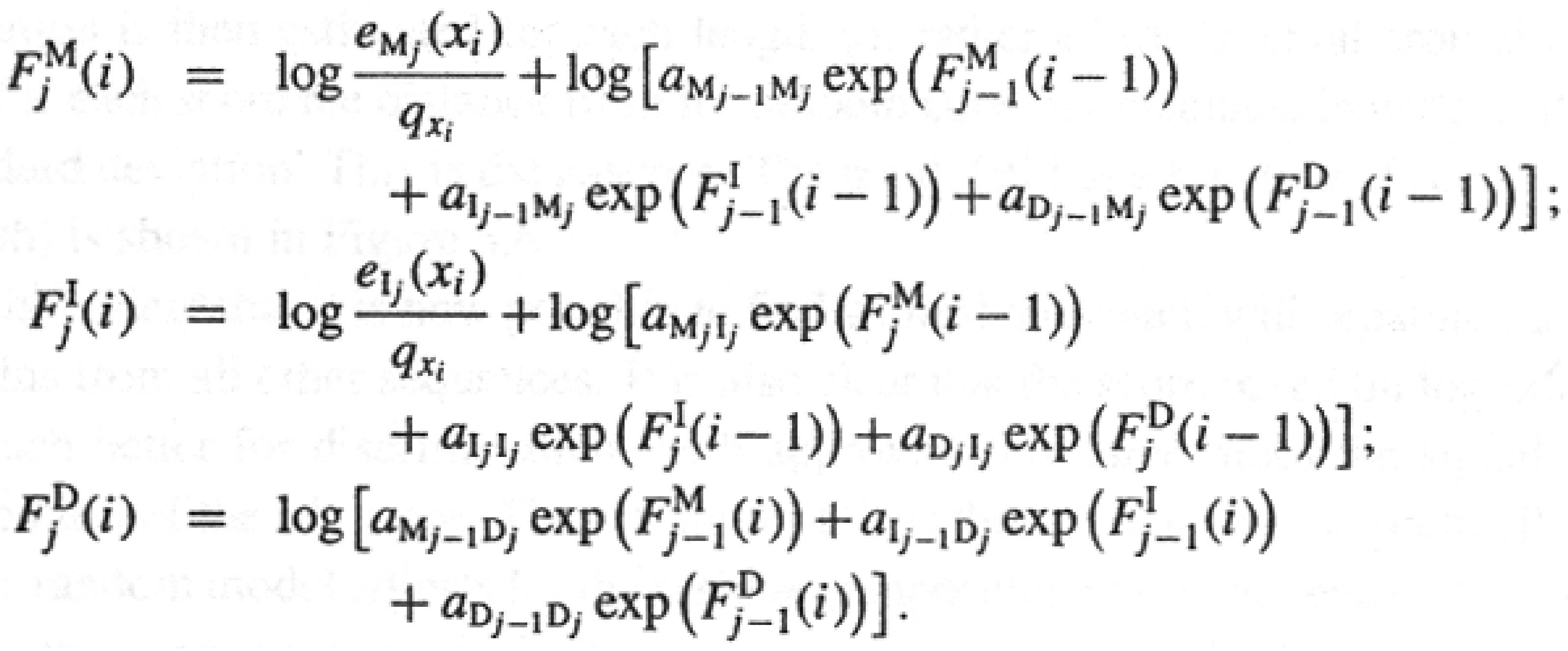


$$
\begin{aligned}
V_j^{\mathrm{M}}(i) &= \log \frac{e_{\mathrm{M}_j}(x_i)}{q_{x_i}} + \max \begin{cases} V_{j-1}^{\mathrm{M}}(i-1) + \log a_{\mathrm{M}_{j-1}\mathrm{M}_j}, \\ V_{j-1}^{\mathrm{I}}(i-1) + \log a_{\mathrm{I}_{j-1}\mathrm{M}_j}, \\ V_{j-1}^{\mathrm{D}}(i-1) + \log a_{\mathrm{D}_{j-1}\mathrm{M}_j}; \end{cases} \\
V_j^{\mathrm{I}}(i) &= \log \frac{e_{\mathrm{I}_j}(x_i)}{q_{x_i}} + \max \begin{cases} V_j^{\mathrm{M}}(i-1) + \log a_{\mathrm{M}_j\mathrm{I}_j}, \\ V_j^{\mathrm{I}}(i-1) + \log a_{\mathrm{I}_j\mathrm{I}_j}, \\ V_j^{\mathrm{D}}(i-1) + \log a_{\mathrm{D}_j\mathrm{I}_j}; \end{cases} \\
V_j^{\mathrm{D}}(i) &= \max \begin{cases} V_{j-1}^{\mathrm{M}}(i) + \log a_{\mathrm{M}_{j-1}\mathrm{D}_j}, \\ V_{j-1}^{\mathrm{I}}(i) + \log a_{\mathrm{I}_{j-1}\mathrm{D}_j}, \\ V_{j-1}^{\mathrm{D}}(i) + \log a_{\mathrm{D}_{j-1}\mathrm{D}_j}. \end{cases}
\end{aligned}
$$

- Initial conditions: $V_0^M(0) = 0$ and $V_0^I = \log \frac{e_{I_0}(x_0)}{q_{x_0}} + \log a_{M_0 I_0}$
- An end state needs to be added
- Similar to NW, only scores are position dependent

Forward algorithm (from Durbin et al.)

- For $s \in \{M, D, I\}$ let $F_j^s(i) = \log \frac{P_M(\boldsymbol{x}_{1:i}, S_{\text{last}}=s_j)}{P_R(\boldsymbol{x}_{1:i})}$

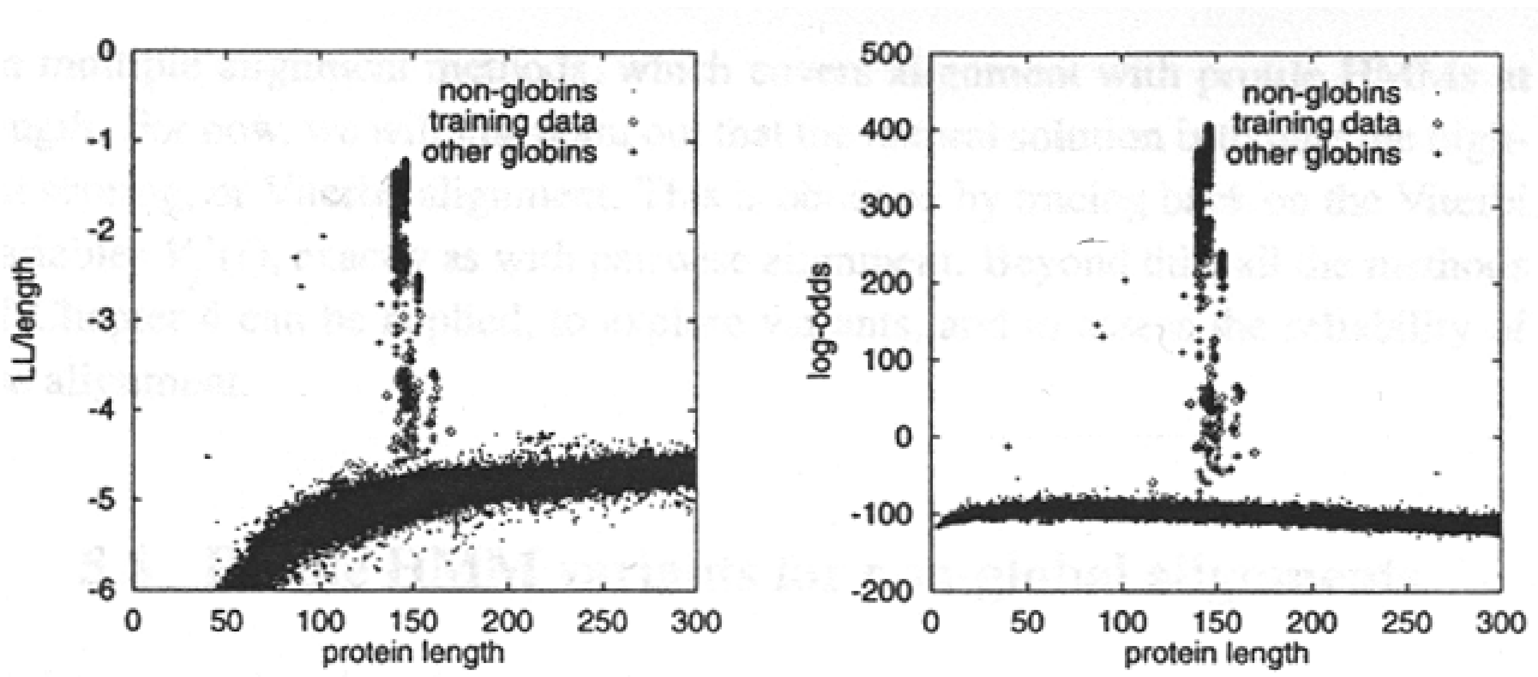


$$\begin{aligned}
F_j^{\mathrm{M}}(i) &= \log \frac{e_{\mathrm{M}_j}(x_i)}{q_{x_i}} + \log\big[a_{\mathrm{M}_{j-1}\mathrm{M}_j} \exp\big(F_{j-1}^{\mathrm{M}}(i-1)\big) \\
&\qquad + a_{\mathrm{I}_{j-1}\mathrm{M}_j} \exp\big(F_{j-1}^{\mathrm{I}}(i-1)\big) + a_{\mathrm{D}_{j-1}\mathrm{M}_j} \exp\big(F_{j-1}^{\mathrm{D}}(i-1)\big)\big]; \\
F_j^{\mathrm{I}}(i) &= \log \frac{e_{\mathrm{I}_j}(x_i)}{q_{x_i}} + \log\big[a_{\mathrm{M}_j\mathrm{I}_j} \exp\big(F_j^{\mathrm{M}}(i-1)\big) \\
&\qquad + a_{\mathrm{I}_j\mathrm{I}_j} \exp\big(F_j^{\mathrm{I}}(i-1)\big) + a_{\mathrm{D}_j\mathrm{I}_j} \exp\big(F_j^{\mathrm{D}}(i-1)\big)\big]; \\
F_j^{\mathrm{D}}(i) &= \log\big[a_{\mathrm{M}_{j-1}\mathrm{D}_j} \exp\big(F_{j-1}^{\mathrm{M}}(i)\big) + a_{\mathrm{I}_{j-1}\mathrm{D}_j} \exp\big(F_{j-1}^{\mathrm{I}}(i)\big) \\
&\qquad + a_{\mathrm{D}_{j-1}\mathrm{D}_j} \exp\big(F_{j-1}^{\mathrm{D}}(i)\big)\big].
\end{aligned}$$

- As before $P_R(\boldsymbol{x}) = \prod_i q_{x_i}$
- $F_0^M(0) = 0$
- $\log(e^x + e^y) = x + \log(1 + e^{y-x})$ and assuming wlog $y < x$ one can use a tabulated $\log(1+h)$ for small h

Example: searching for globins

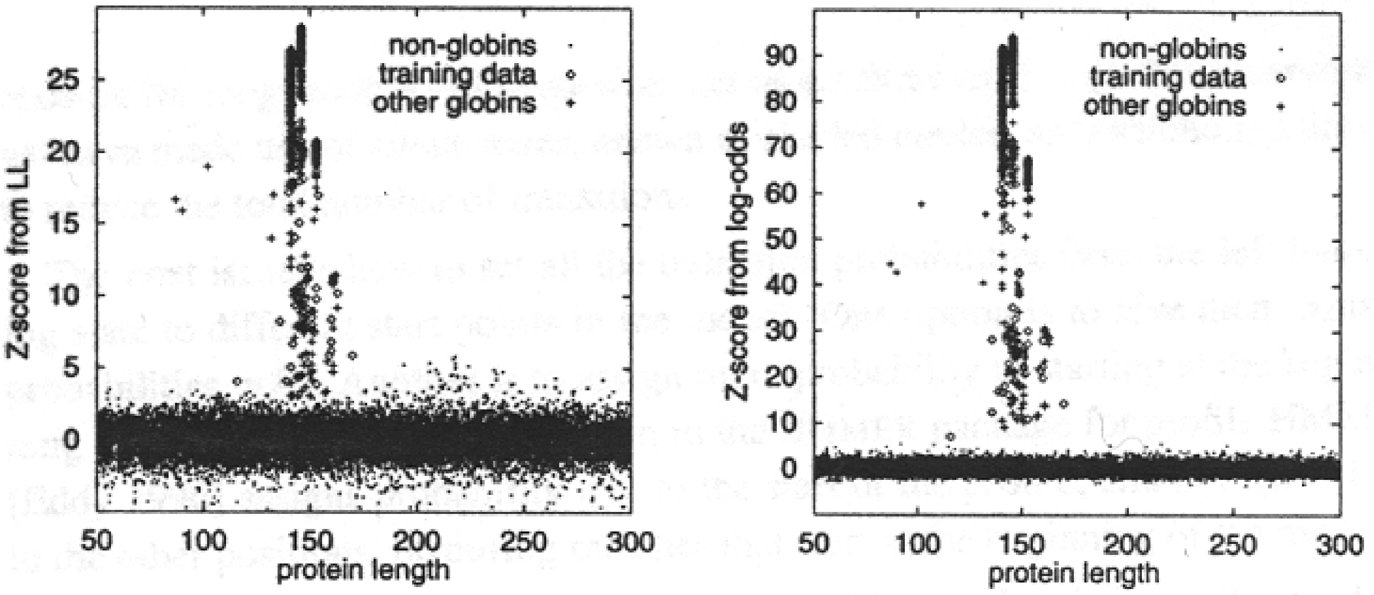
- 300 randomly picked globin sequences generated profile HMM
- SWISS-PROT (r.34) which contained $\sim 60,000$ proteins was searched
  - using the forward algorithm for computing both LL and LLR
    - ▷ the null model was generated from the trainning set



- Note the difference in the variance and normalization problems

- Choosing a cutoff of 0 for the LLR will lead to many false negatives:
  - the training set is not sufficiently diverse

- Can use Z-scores to fix that:
  - fit a smooth "average" curve to each of the non-globins graphs
  - estimate a "local" standard deviation (use a small window)
  - replace each score s_i by $\frac{s_i - \mu(l_i)}{\sigma(l_i)}$



  - LLR is a better predictor: without normalizing sequences with a similar composition to globins tend to score higher

Finding the average curve - moving average

- The data is modeled as random fluctuations about a determinstic curve
- The original approach by Krogh et al. (94) used windows of roughly 500 non-globin sequences of similar length
- The scores and lengths in each window were averaged
- The average curve is the piecewise linear curve connecting the averages
- Linear regression was used in the first and last windows
- Standard deviations are computed per window
- Remove outliers, re-estimate average curve and iterate
- This is a slight modification of the moving average method

Finding the average curve - LOWESS and LOESS

- LOWESS and LOESS (Cleveland 79,88) - locally weighted regression and smoothing scatter plot
  - use locally weighted polynomial regression to smooth data
    - ▷ or, build the deterministic part of the variation in the data
- At each point (length) x_0 of the data consider only the data in N_{x_0}, a local neighborhood of fixed size about x_0
  - regress data in N_{x_0} on first (LOWESS) or second (LOESS) degree polynomials
  - use weighted regression, with $d := d(x_0) := \max_{x \in N_{x_0}} |x - x_0|$

tri-cube: $$w(x) = \begin{cases} \left[1 - \left(\frac{x - x_0}{d}\right)^3\right]^3 & |x - x_0| < d \\ 0 & |x - x_0| \geq d \end{cases}$$

  - Weighted regression: find $\min_f \sum_i w_i |y_i - f(x_i)|^2$