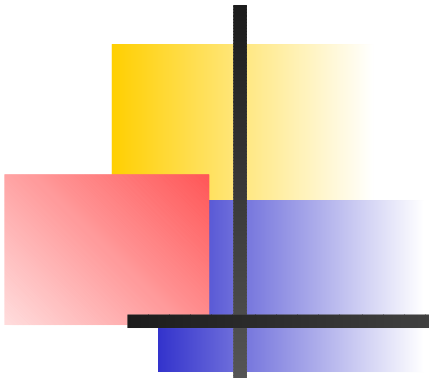


Phylogenetic Inference: Sequence Alignment

Philippe Lemey and Marc A. Suchard

Rega Institute

Department of Microbiology and Immunology

K.U. Leuven, Belgium, and

Departments of Biomathematics and Human Genetics

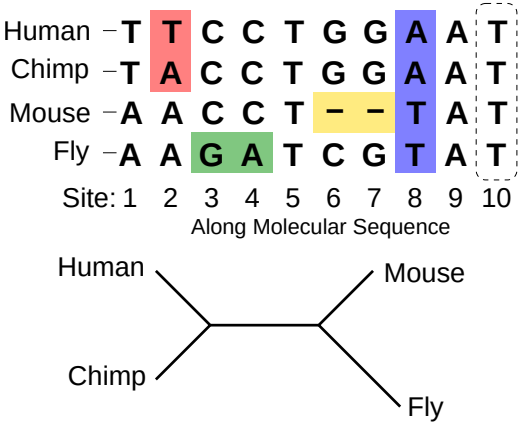
David Geffen School of Medicine at UCLA

Department of Biostatistics

UCLA School of Public Health

Sequence Alignments

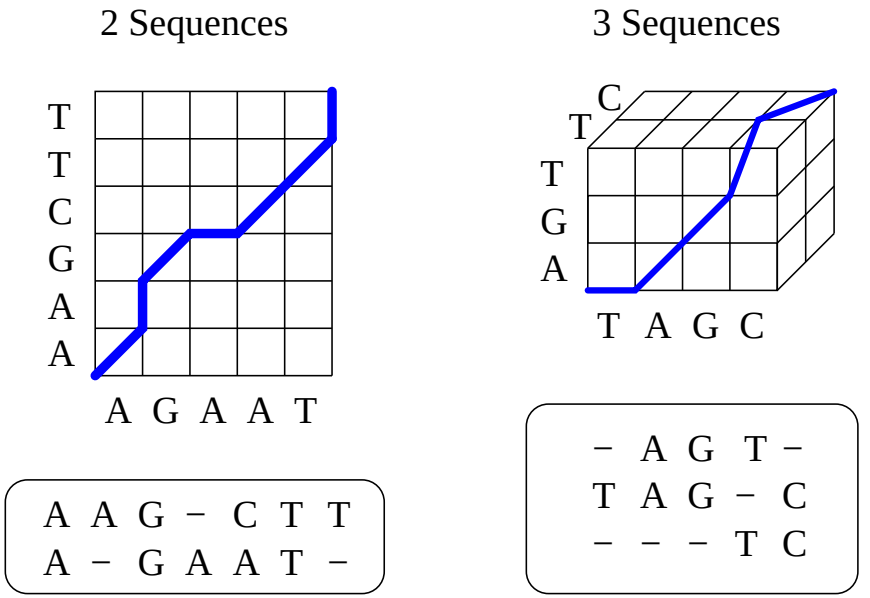
Assign homology between characters from different taxa. Two characters are **homologous** if they share a common ancestor by vertical descent.



Important:

- Homology statements allow one to find conserved (functionally important) sites
- Almost universally, phylogenetic methods condition on (**assume as fixed**) a sequence alignment. Concern: "garbage-in, garbage-out"

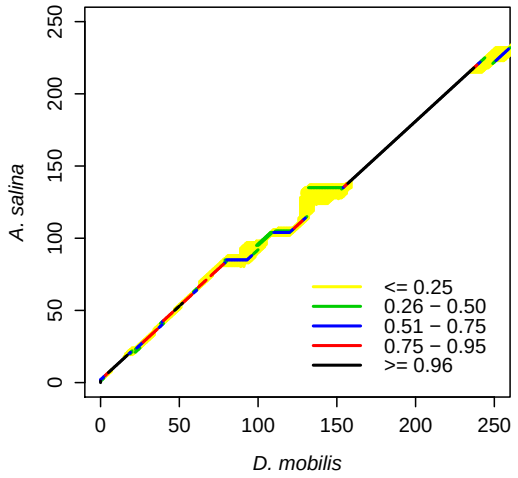
Alignment Path Graphs

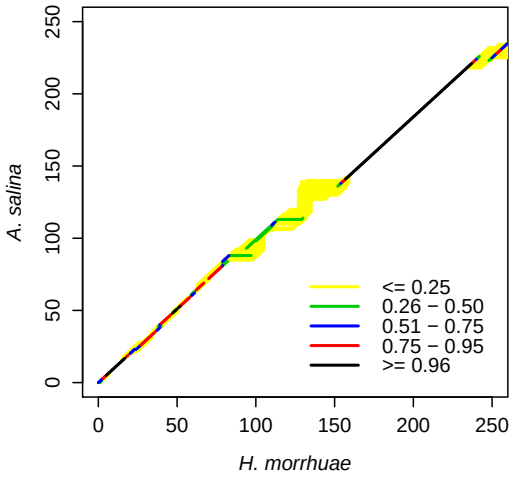


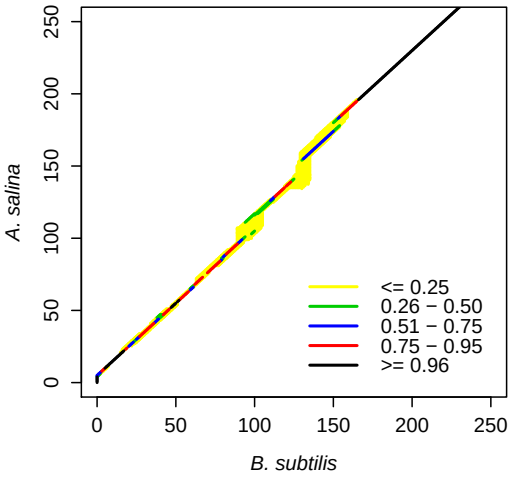
Represent a multiple alignment as an increasing **path** within a N-dimensional lattice cube:

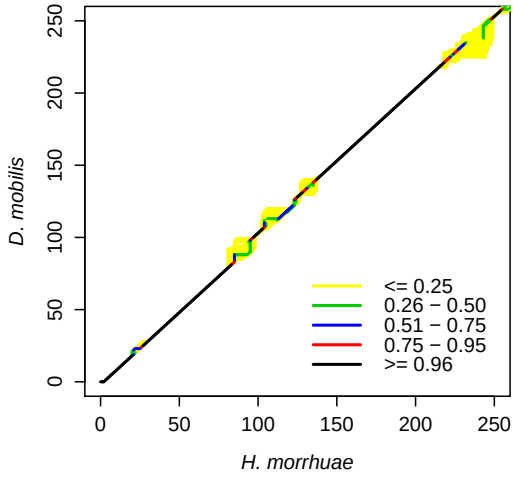
- Runs from lattice point $(0, \ldots, 0)$ to $(\ell_1, \ldots, \ell_N)$, and
- Edges (segments between points) correspond to alignment patterns.

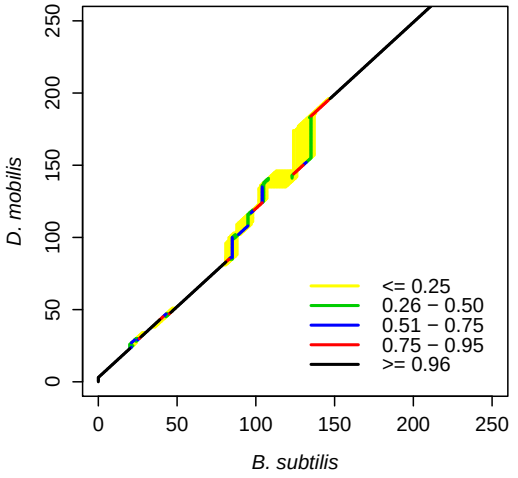
16/18S rRNA Alignment for the Tree of Life

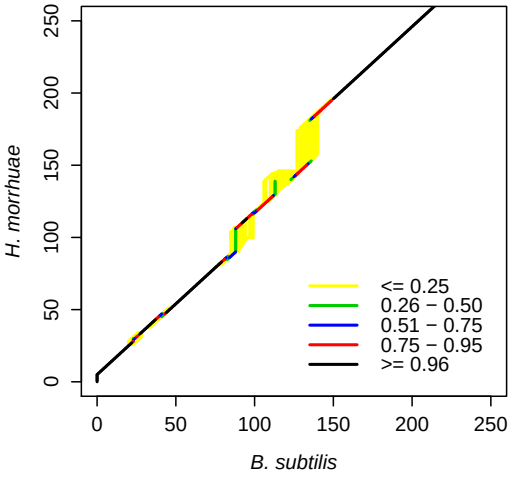






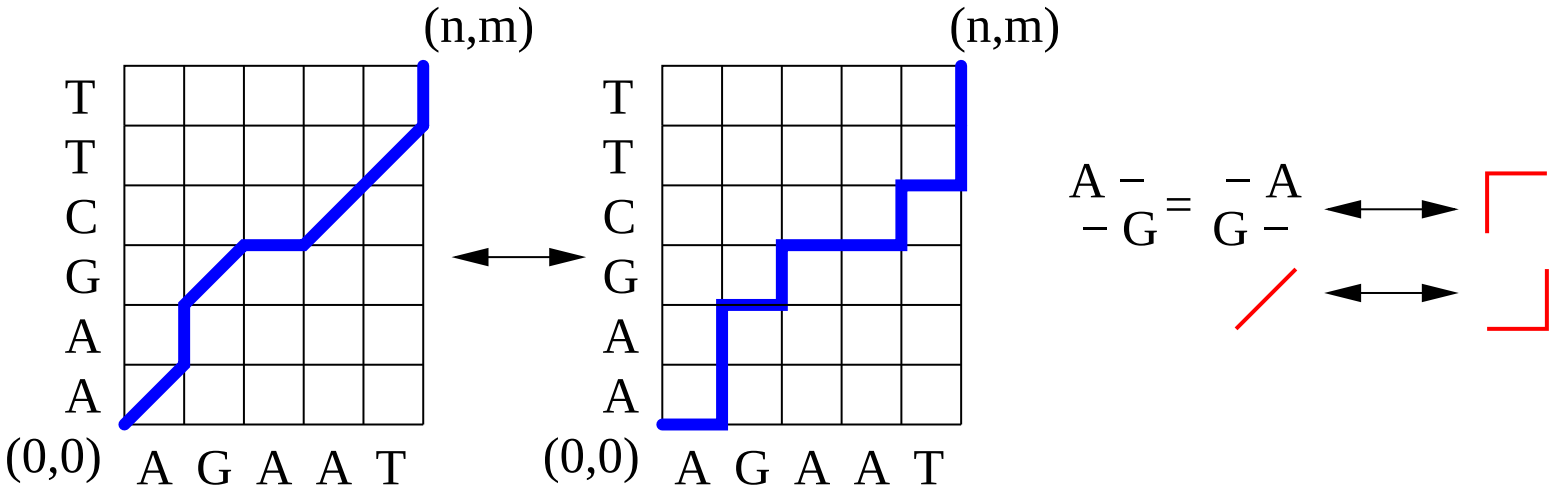






Counting Pairwise (2-Taxon) Alignments

Equivalence between alignment and lattice path graphs



Alignment Path **Lattice Path**

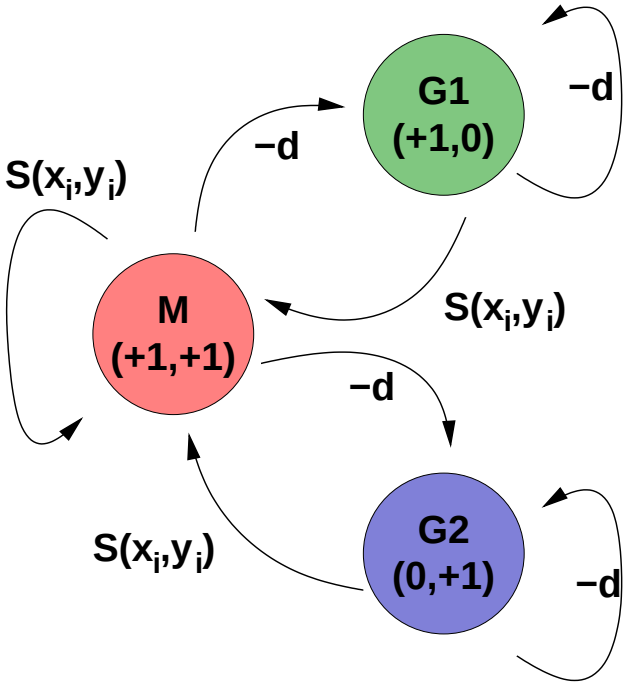
Impose a strict ordering on gaps. Then there exist $\binom{n+m}{n}$ possible lattice paths

- Justification: there are $n + m$ total steps of which n must be to the right

Modeling Pairwise Alignments

Consider a Markov chain on the following graph:

- 3 states – match (M) and gaps (G1/G2)
- characters x_i and y_i are homologous with score $S(x_i, y_i)$
- a gap scores $-d$



Extensions:

- Gap opening vs. gap elongation
- Fix (BLOSUM50/$d = 8$) or estimate scores

Finding the Optimal Path

Use dynamic programming (or forward-backward algorithm) to reduce exponential search space to polynomial $O(nm)$

Needleman-Wunsch algorithm:

- Idea: build up optimal alignment from previous solutions for smaller subsequences
- How: Construct matrix $\boldsymbol{F} = \{F(i,j)\}$, where $F(i,j)$ equals the **score** of the optimal alignment between segments $x_1 \cdots x_i$ and $y_1 \cdots y_j$
- Key: $\boldsymbol{F}$ is build recursively

	A	**G**	**C**	**T**
A	10	−1	−3	−4
G	−1	7	−5	−3
C	−3	−4	9	0
T	−4	−3	0	8

Example $S(x_i, y_j)$

Needleman-Wunsch algorithm

$F(0,0) \leftarrow 0, F(i,0) \leftarrow -id, F(0,j) \leftarrow -j \; i = 1,\ldots,N, \; j = 1,\ldots,M$ **{initialization}**

for $i = 1,\ldots,N$ **do**

  for $j = 1,\ldots,M$ **do**

    $f_{\mathsf{M}} \leftarrow F(i-1,j-1) + S(x_i, y_j)$

    $f_{\mathsf{G1}} \leftarrow F(i-1,j) - d$

    $f_{\mathsf{G2}} \leftarrow F(i,j-1) - d$

    $F(i,j) = \max\{f_{\mathsf{M}}, f_{\mathsf{G1}}, f_{\mathsf{G2}}\}$ {always chose the optimal}

    Label $F(i,j)$ with a pointer to the previous M, G1 or G2 selected cell

  end for

end for

$F(n,m)$ is the optimal score

Follow pointers from $F(n,m) \rightarrow F(0,0)$ to obtain optimal alignment **{traceback}**

Intuition: nested $\max$ statements leads to a "branch-and-bound"-like path

Model Pairwise Alignment as a Hidden Markov Model

Permits simultaneous estimation of HMM parameters θ

Work-horses:

- **Forward-backward algorithm** – computes the probability of the alignment given θ
- **Baum-Welch algorithm** – computes ML estimates (posterior modes) of θ given the alignment
- **Viterbi algorithm** – finds the most likely sequence of hidden states

More details in course if time permits