Multiple Sequence Alignment and Profiles

Dr. Fayyaz ul Amir Afsar Minhas

PIEAS Biomedical Informatics Research Lab

Department of Computer and Information Sciences

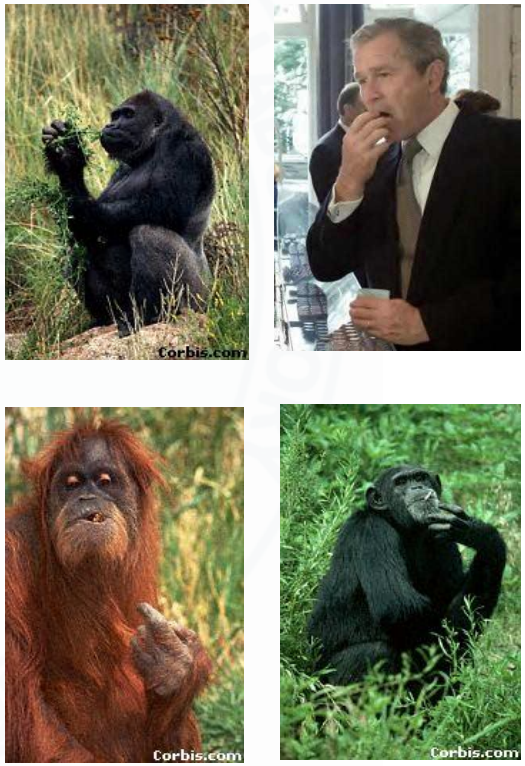
Pakistan Institute of Engineering & Applied Sciences

PO Nilore, Islamabad, Pakistan

http://faculty.pieas.edu.pk/fayyaz/

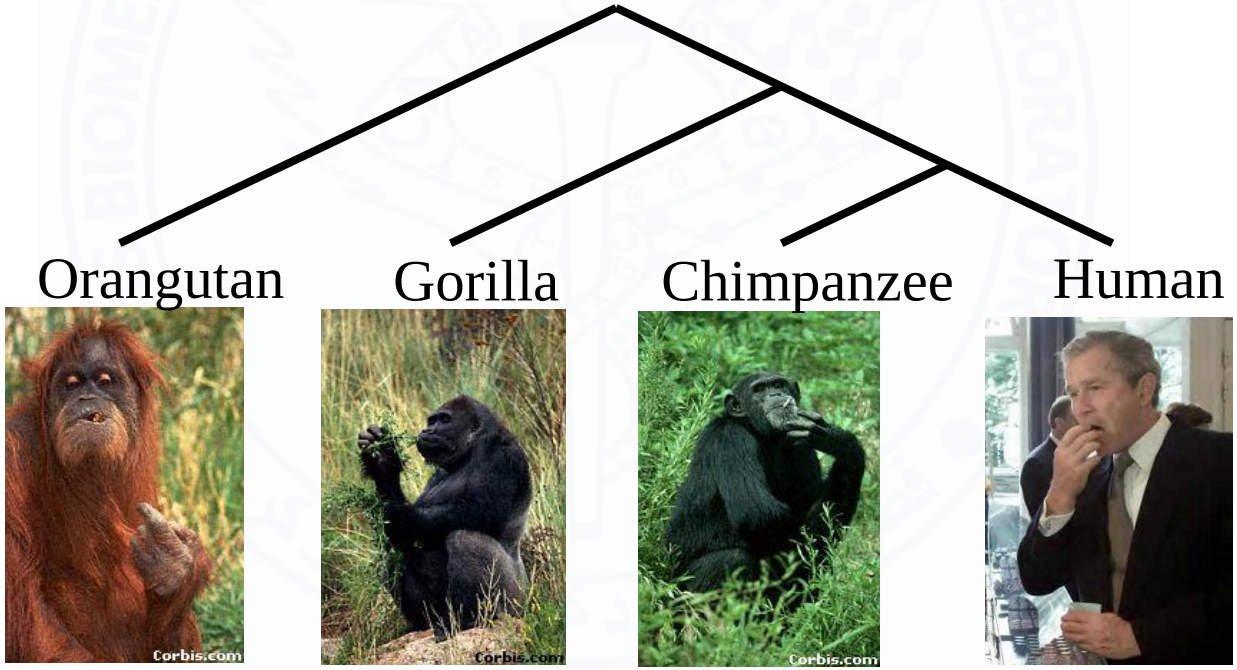
Multiple Alignmet vs. Pairwise Alignments

- Up until now we have only tried to align two sequences.
- What about more than two?
  - And what for?



What for?

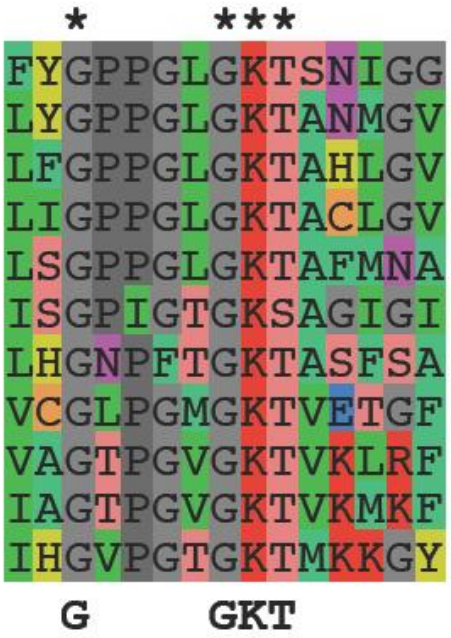
- Many, Many uses…
  - Establishing the relationships between species

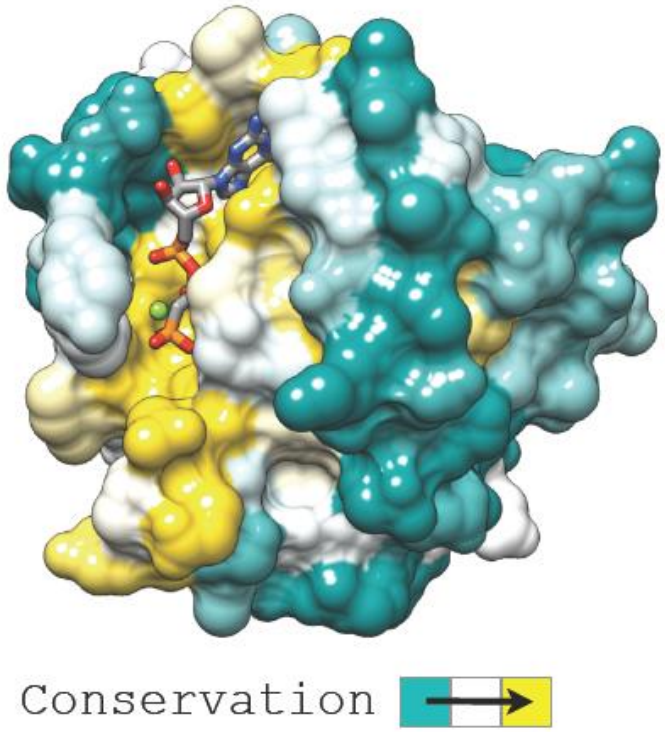


What for?

- Identifying regions in sequences that have important biological roles
  - Motifs

ATP/GTP-binding proteins: G-x(4)-G-K-T

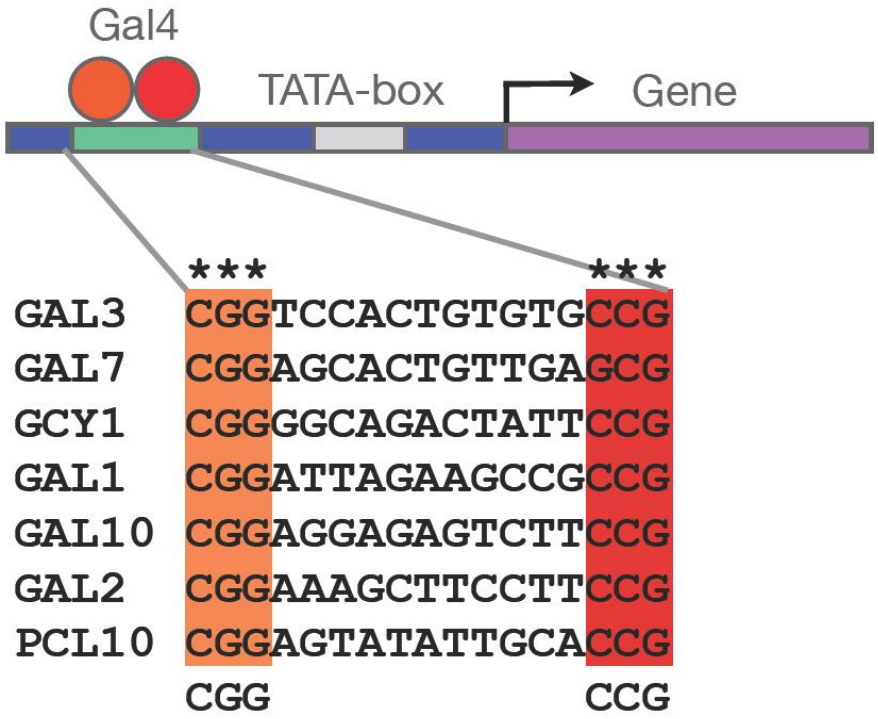




Functional cues from conservation patterns...

Many DNA patterns are binding sites for Transcription Factors.

- E.g., The Gal4 binding sequence
  C-G-G-N(11)-C-C-G



Motifs

- Motifs are sequence patterns common to a lot of biological sequences and have biological significance
- NF-B Binding site
  - All ‘immunity’ genes have a pattern located upstream of the start of the genes
  - TCGGGGATTTCC
  - Examples of regulatory motifs that turn on immunity and other genes
  - Allow proteins called transcription factors to bind here which recruits RNA polymerase
- CAM Binding site (IQ Motif)
  - [FILV]Qxxx[RK]Gxxx[RK]xx[FILVWY]
  - Usually starts with IQ so the consensus sequence is IQxxxRGxxxR

PROSITE is a protein pattern and profile database

Currently contains > 1600 patterns and profiles: http://prosite.expasy.org/
Example PROSITE patterns:

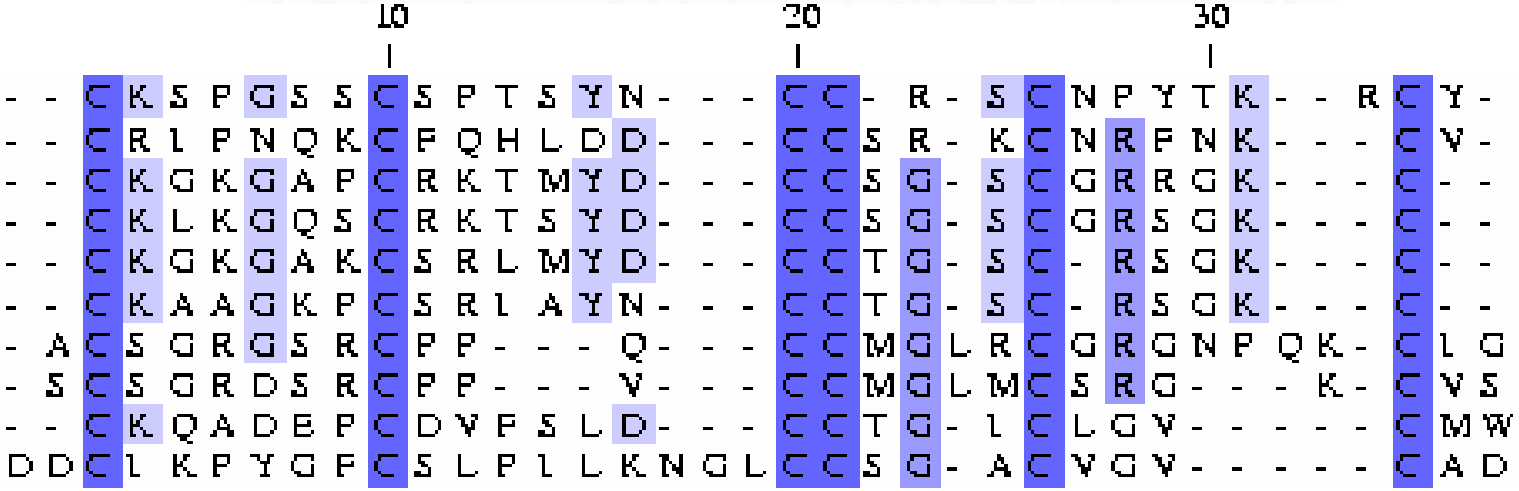
PS00087; SOD_CU_ZN_1
[GA]-[IMFAT]-H-[LIVF]-H-{S}-x-[GP]-[SDG]-x-[STAGDE]
The two Histidines are copper ligands

- Each position in pattern is separated with a hyphen
- x can match any residue
- [ ] are used to indicate ambiguous positions in the pattern
  e.g., [SDG] means the pattern can match S, D, or G at this position
- { } are used to indicate residues that are not allowed at this position
  e.g., {S} means NOT S (not Serine)
- ( ) surround repeated residues, e.g., A(3) means AAA

Information from http://ca.expasy.org/prosite/prosuser.html

Correctness of an alignment

- Homologous residues should be in the same column of an MSA.



Generalizing the Notion of Pairwise Alignment

- Alignment of 2 sequences: a 2-row matrix
- Alignment of 3 sequences: a 3-row matrix

```
A T - G C G -
A - C G T - A
A T C A C - A
```

- Score: more conserved columns, better alignment

Alignment = Paths in…

Align 3 sequences: ATGC, AATC, ATGC

	A	--	T	G	C

	A	A	T	--	C

	--	A	T	G	C

Alignment Paths

0	1	1	2	3	4
	A	--	T	G	C

	A	A	T	--	C

	--	A	T	G	C

x coordinate

Alignment Paths

0	1	1	2	3	4
	A	--	T	G	C

x coordinate

0	1	2	3	3	4
	A	A	T	--	C

y coordinate

	--	A	T	G	C

Alignment Paths

0	1	1	2	3	4
	A	--	T	G	C
0	1	2	3	3	4
	A	A	T	--	C
0	0	1	2	3	4
	--	A	T	G	C

x coordinate

y coordinate

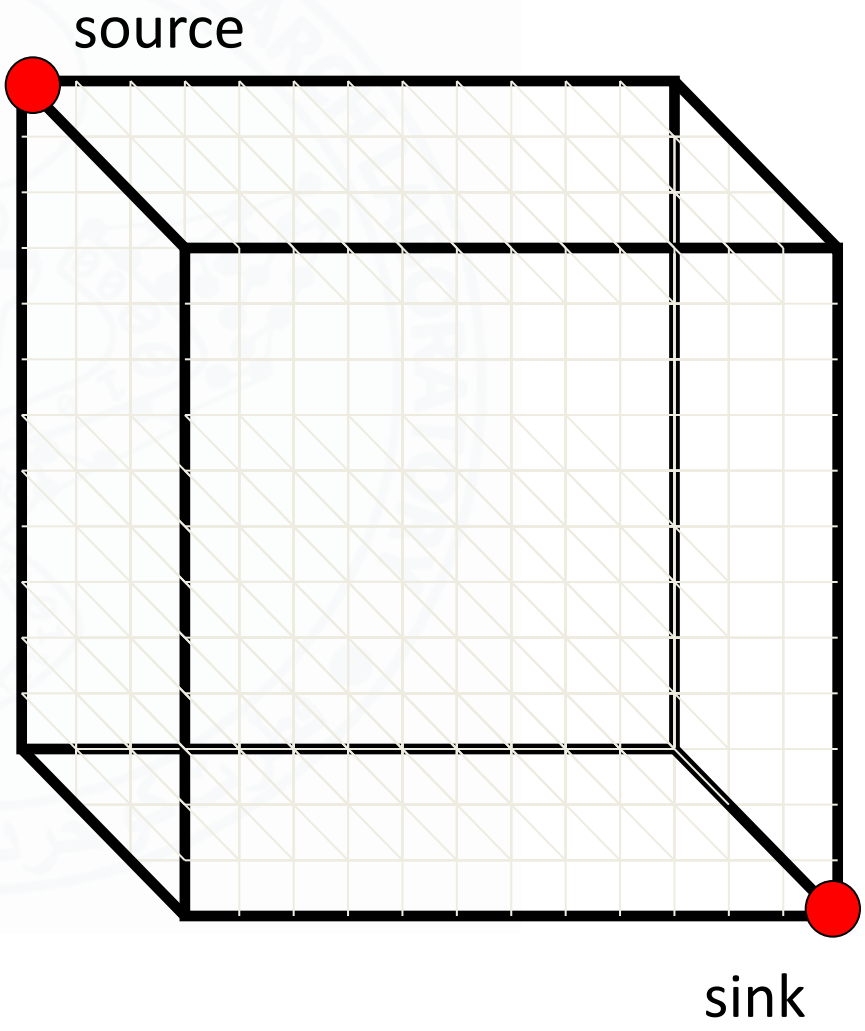
z coordinate

- Resulting path in *(x,y,z)* space:

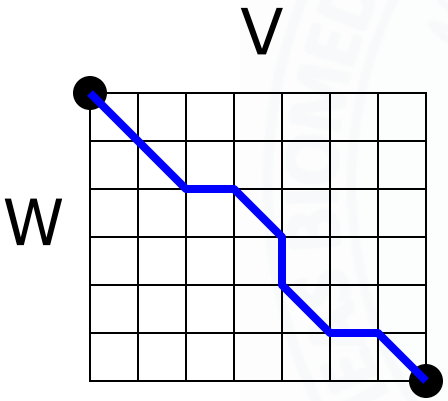
$$(0,0,0)\rightarrow(1,1,0)\rightarrow(1,2,1)\rightarrow(2,3,2)\rightarrow(3,3,3)\rightarrow(4,4,4)$$

Aligning Three Sequences

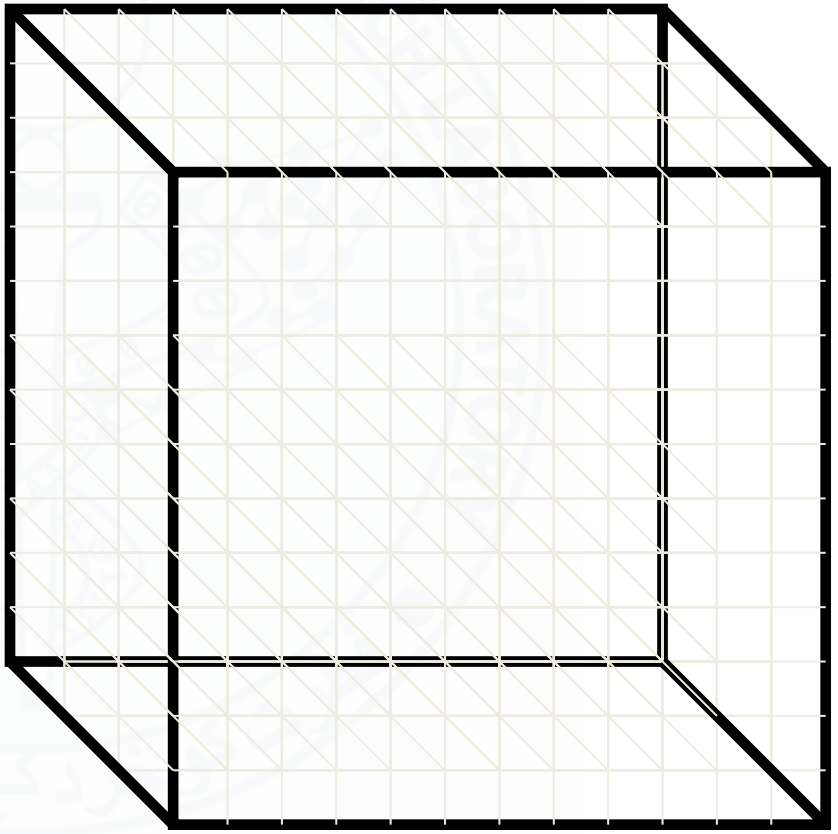
- Same strategy as aligning two sequences
- Use a 3-D grid, with each axis representing a sequence to align
- For global alignment, go from source to sink



2-D vs 3-D Alignment Grid

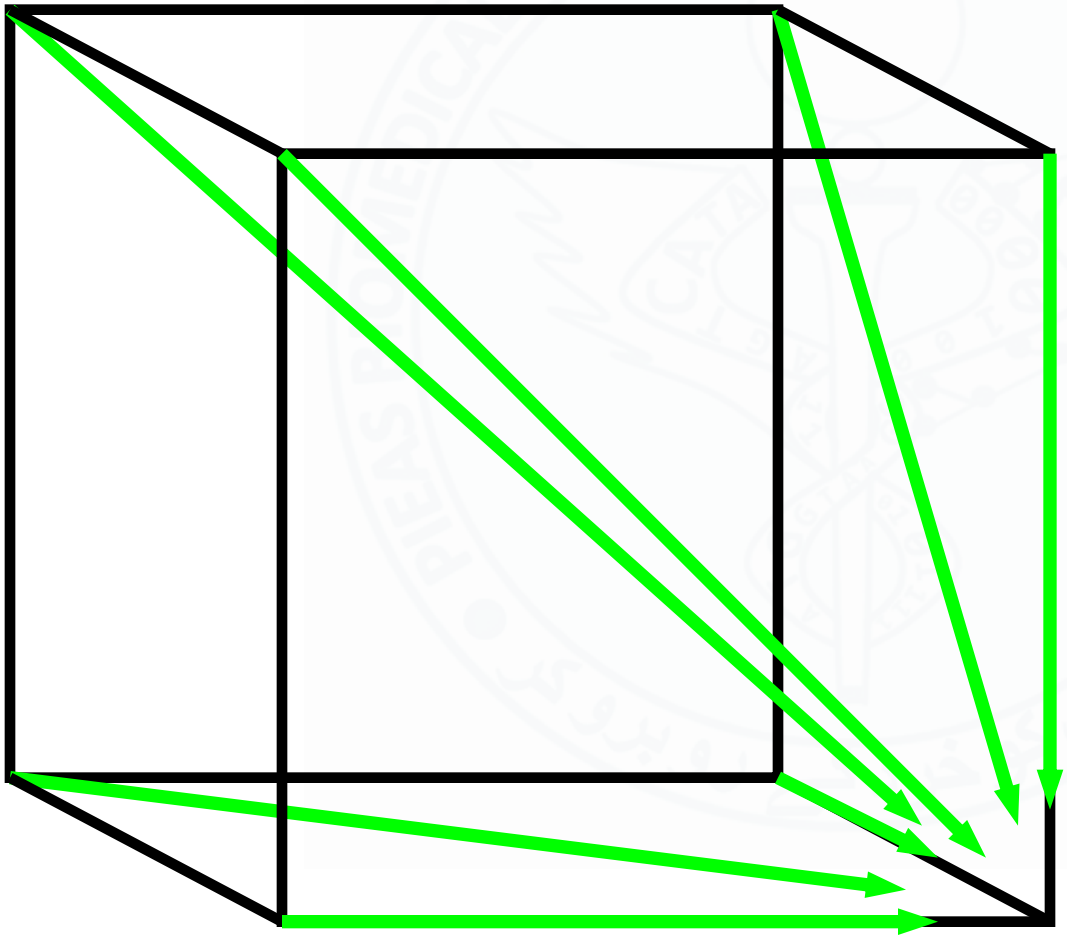


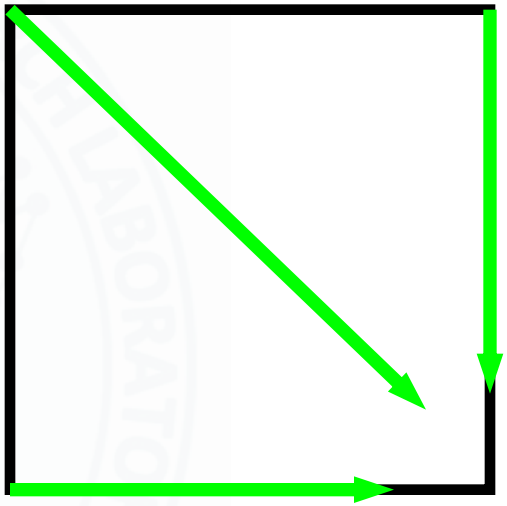
2-D edit graph



3-D edit graph

2-D cell versus 3-D Alignment Cell

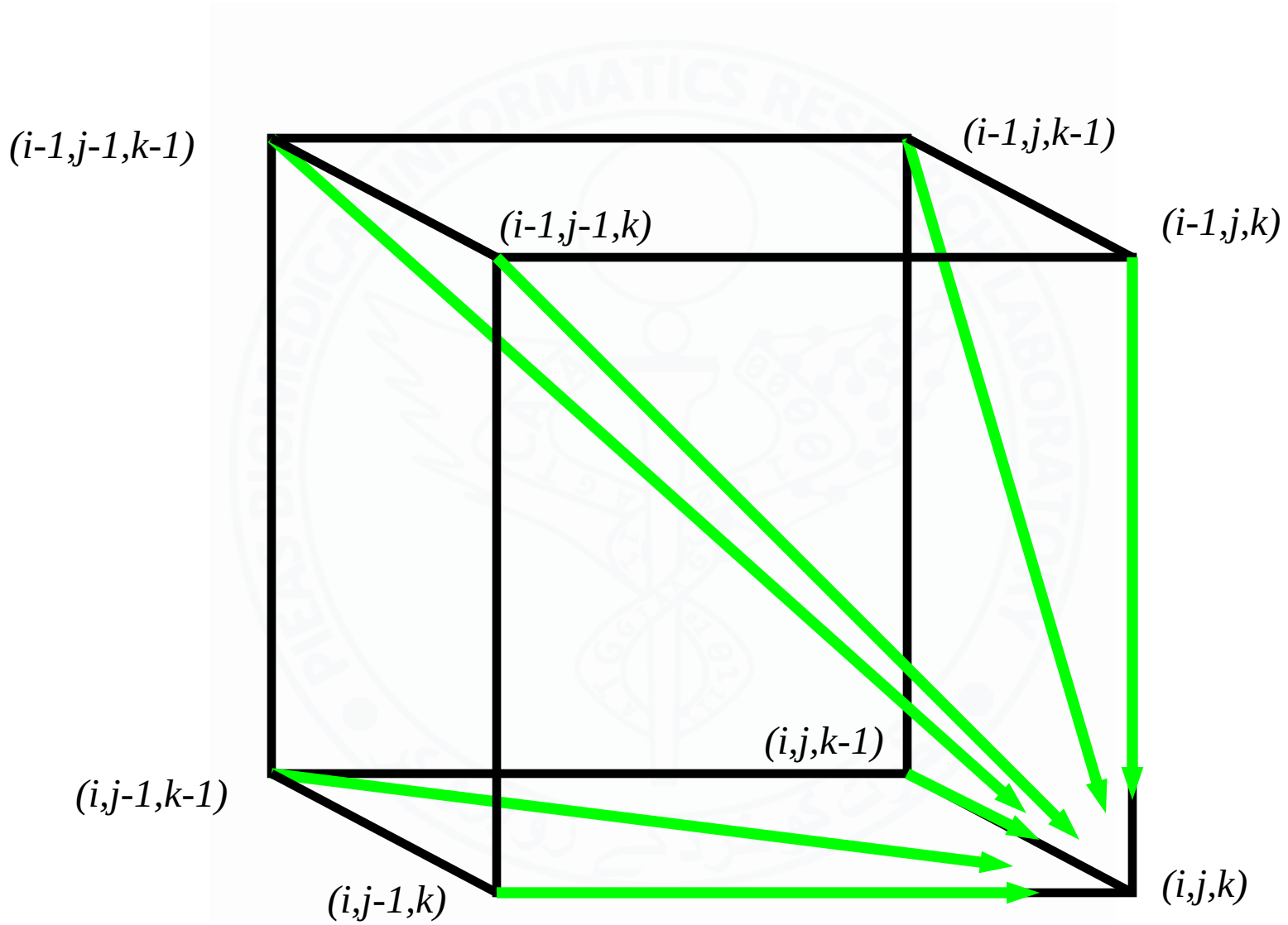




In 2-D, 3 edges in each unit square

In 3-D, 7 edges in each unit cube

Architecture of 3-D Alignment Cell



Multiple Alignment: Dynamic Programming

$$s_{i,j,k} = \max \begin{cases} s_{i-1,j-1,k-1} + \delta(v_i, w_j, u_k) \\ s_{i-1,j-1,k} + \delta(v_i, w_j, \_) \\ s_{i-1,j,k-1} + \delta(v_i, \_, u_k) \\ s_{i,j-1,k-1} + \delta(\_, w_j, u_k) \\ s_{i-1,j,k} + \delta(v_i, \_, \_) \\ s_{i,j-1,k} + \delta(\_, w_j, \_) \\ s_{i,j,k-1} + \delta(\_, \_, u_k) \end{cases}$$

cube diagonal: no indels

face diagonal: one indel

edge diagonal: two indels

$\delta(x, y, z)$ is an entry in the 3-D scoring matrix

Multiple Alignment: Running Time

- For 3 sequences of length *n*, the run time is $7n^3$: O(n^3)
- Running time for k sequences?

Multiple Alignment: Running Time

- For 3 sequences of length *n*, the running time is $O(n^3)$
- For *k* sequences: k-dimensional grid. Running time: $(2^k-1)(n^k)$ i.e. $O(2^k n^k)$
- Conclusion: dynamic programming approach is easily extended to *k* sequences but is impractical.

What next?

- Heuristics: reducing MSA to pairwise alignment.

Reducing MSA to pairwise alignment

- **Progressive alignment** - a succession of pairwise alignments: at each step align sequence to an MSA that was already computed
- The primary heuristic: start with the most similar pairs of sequences since these will produce the most reliable alignments.
- Pro: Fast
- Con: What are we optimizing?

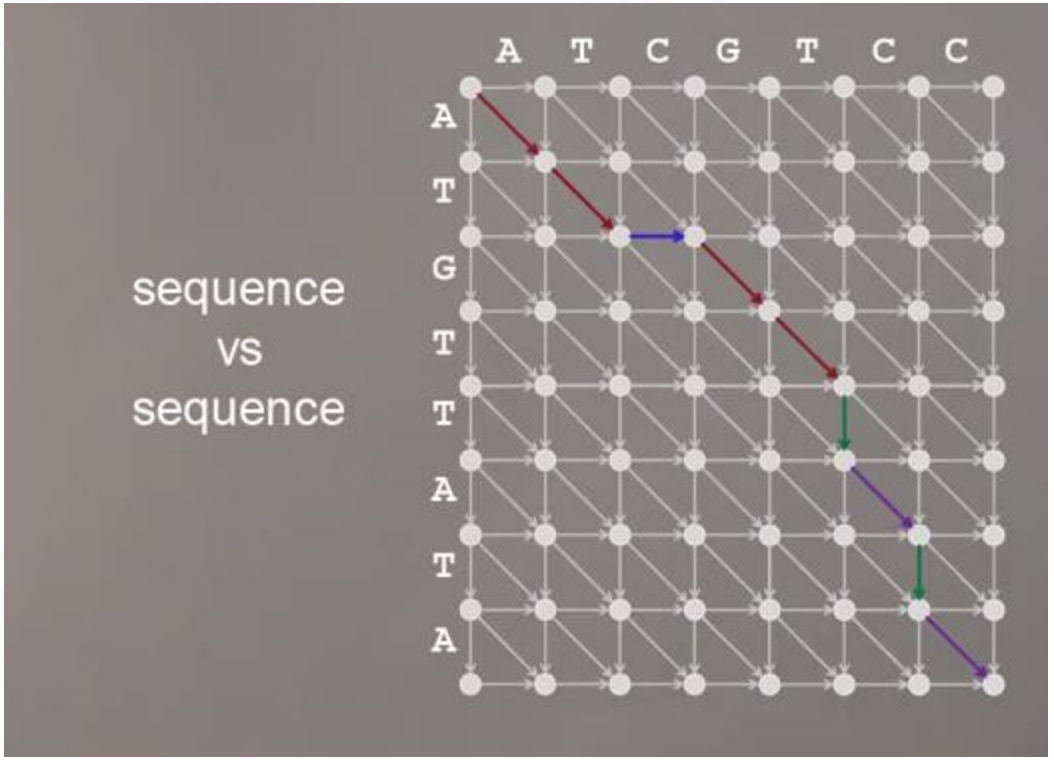
Profile Representation of an MSA

	-	A	G	G	C	T	A	T	C	A	C	C	T	G
	T	A	G	-	C	T	A	C	C	A	-	-	-	G
	C	A	G	-	C	T	A	C	C	A	-	-	-	G
	C	A	G	-	C	T	A	T	C	A	C	-	G	G
	C	A	G	-	C	T	A	T	C	G	C	-	G	G
A	0	1	0	0	0	0	1	0	0	.8	0	0	0	0
C	.6	0	0	0	1	0	0	.4	1	0	.6	.2	0	0
G	0	0	1	.2	0	0	0	0	0	.2	0	0	.4	1
T	.2	0	0	0	0	1	0	.6	0	0	0	0	.2	0
-	.2	0	0	.8	0	0	0	0	0	0	.4	.8	.4	0

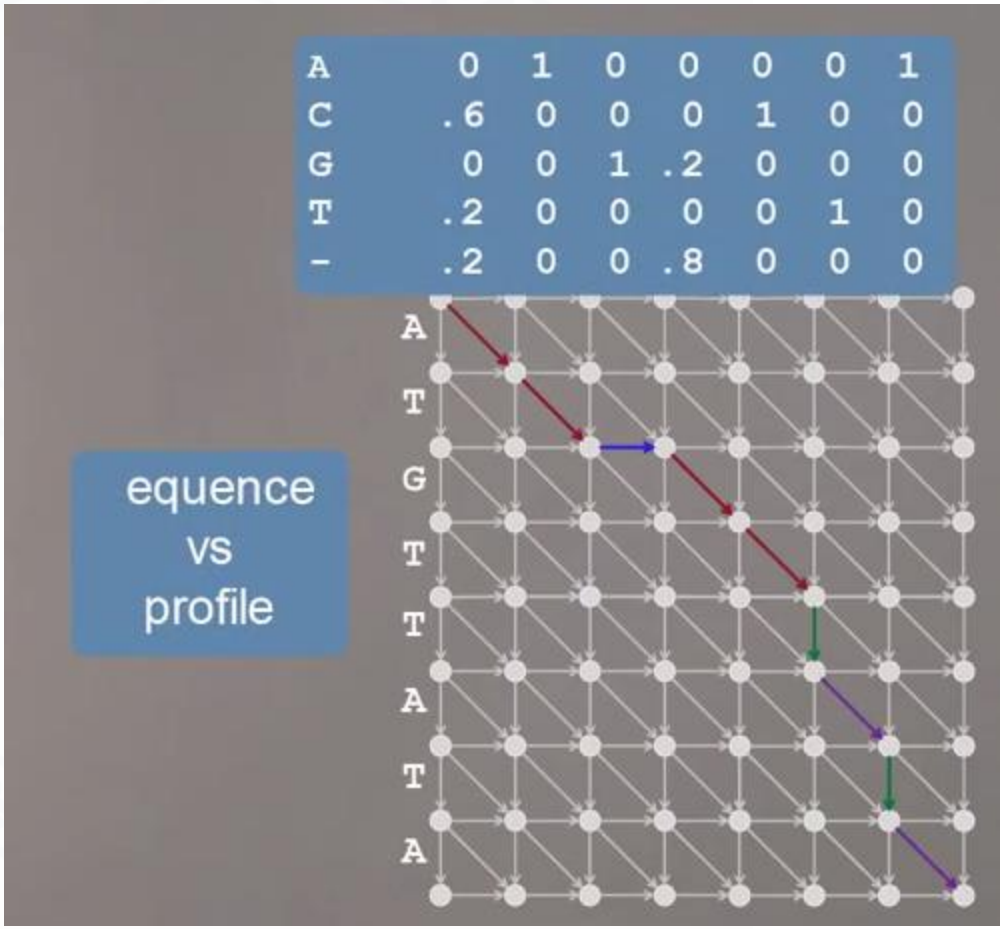
Profile alignment

- How to align a sequence against a profile?
- How to align a profile against a profile?
- An alignment of two profiles induces a multiple sequence alignment of the sequences

Sequence vs. Sequence



Sequence vs. Profile



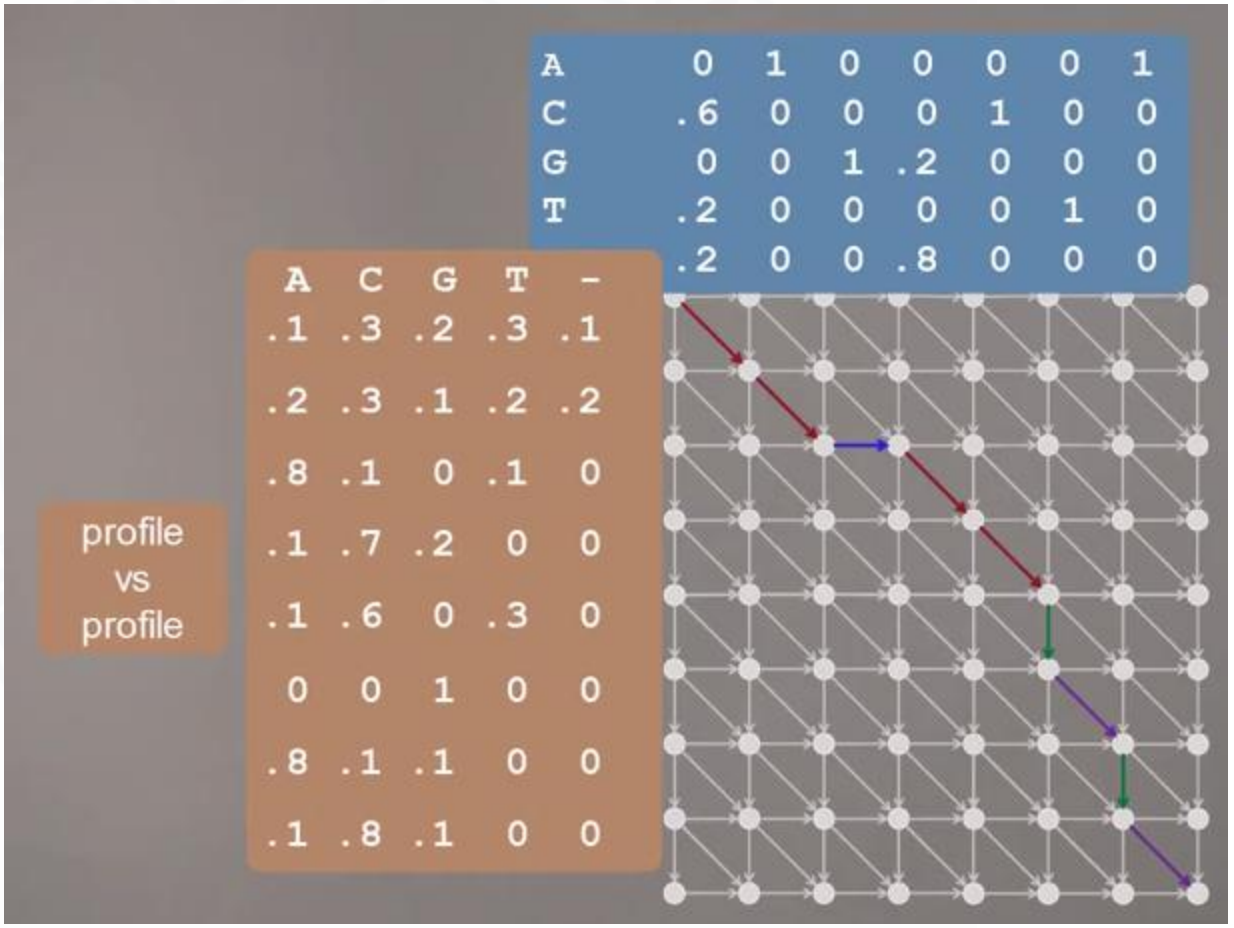
- The score the profile for amino acid a at position p is

$$M(p,a) = \sum_{b=1}^{20} f(p,b) \cdot s(a,b)$$

where

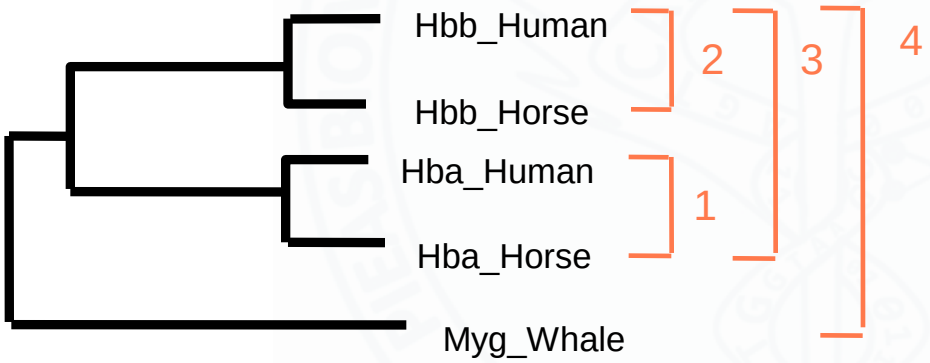
  - f(p,b) = frequency of amino acid b in position p
  - s(a,b) is the score of (a,b) (from, e.g., BLOSUM or PAM)

Profile vs. Profile

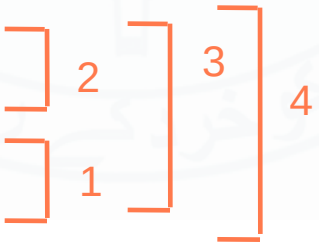


Overview of ClustalW

```
Hbb_Human   1   -
Hbb_Horse   2   .17   -
Hba_Human   3   .59   .60   -
Hba_Horse   4   .59   .59   .13   -
Myg_Whale   5   .77   .77   .75   .75   -
```




```
1   PEEKSAVTALWGKVN--VDEVGG
2   GEEKAAVLALWDKVN--EEEVGG
3   PADKTNVKAAWGKVGAHAGEYGA
4   AADKTNVKAAWSKVGGHAGEYGA
5   EHEWQLVLHVWAKVEADVAGHGQ
```



sequences

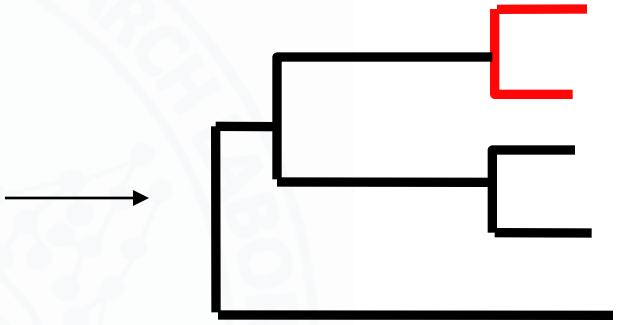
distance matrix

Guide tree
(Neighbor-joining)

Progressive alignment
following guide tree

Progressive alignment - step 1

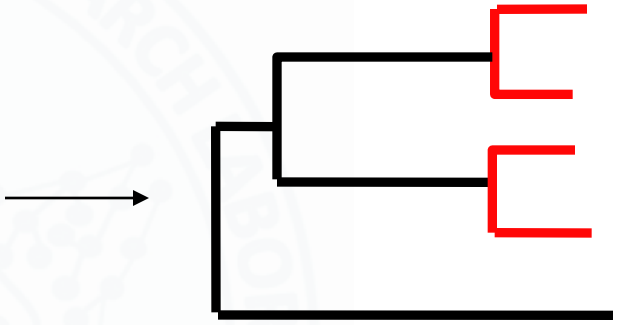
gctcgatacgatacgatgactagcta
gctcgatacaagacgatgacagcta
gctcgatacacgatgactagcta
gctcgatacacgatgacgagcga
ctcgaacgatacgatgactagct



gctcgatacgatacgatgactagcta
gctcgatacaagacgatgac-agcta

Progressive alignment - step 2

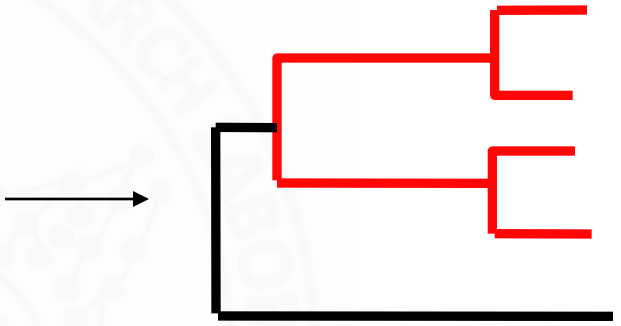
gctcgatacgatacgatgactagcta
gctcgatacaagacgatgacagcta
gctcgatacacgatgactagcta
gctcgatacacgatgacgagcga
ctcgaacgatacgatgactagct



gctcgatacacgatgactagcta
gctcgatacacgatgacgagcga

Progressive alignment - step 3

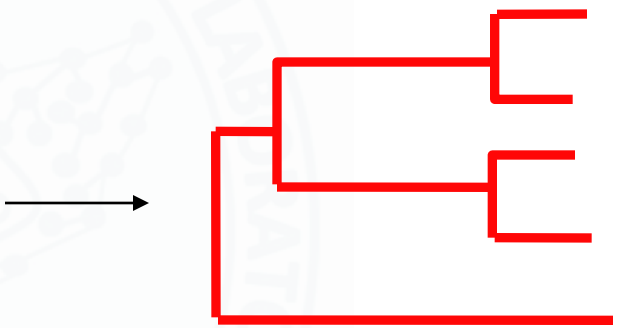
```
gctcgatacgatacgatgactagcta
gctcgatacaagacgatgac-agcta
+
gctcgatacacgatgactagcta
gctcgatacacgatgacgagcga
```



```
   gctcgatacgatacgatgactagcta
   gctcgatacaagacgatgac-agcta
   gctcgatacacga---tgactagcta
   gctcgatacacga---tgacgagcga
```

Progressive alignment - final step

```
gctcgatacgatacgatgactagcta
gctcgatacaagacgatgac-agcta
gctcgatacacga---tgactagcta
gctcgatacacga---tgacgagcga
+
ctcgaacgatacgatgactagct
```



```
gctcgatacgatacgatgactagcta
gctcgatacaagacgatgac-agcta
gctcgatacacga---tgactagcta
gctcgatacacga---tgacgagcga
-ctcga-acgatacgatgactagct-
```

Progressive alignment

- Many flavors of progressive alignment.
- Algorithms differ in:
  - The order in which sequences are aligned (guide tree/no guide tree/what type of guide tree)
  - Scoring and alignment of a sequence/alignment to an alignment.

Where things can go wrong…

SequenceA GARFIELD THE LAST FAT CAT
SequenceB GARFIELD THE FAST CAT
SequenceC GARFIELD THE VERY FAST CAT
SequenceD THE FAT CAT

Clustal alignment

```
Sequence A GARFIELD THE LAST FA-T CAT
Sequence B GARFIELD THE FAST CA-T ---
Sequence C GARFIELD THE VERY FAST CAT
Sequence D -------- THE ---- FA-T CAT
```

Another example

A problem of progressive alignment:

- Initial alignments are “frozen” even when new evidence comes

Example:

```
x:    GAAGTT
y:    GAC-TT

z:    GAACTG
w:    GTACTG
```

Frozen!

Now clear that y = GA- CTT is correct

Iterative Refinement

- Barton-Sternberg method:

Create an alignment using a progressive method.

for *i=1,…,N*

Remove sequence i and realign it to a profile of the other aligned sequences.

Repeat until convergence

MUSCLE

Combines several ideas:

- Draft progressive alignment
- Recompute a guide tree and recompute a progressive alignment.
- Refinement

MUSCLE

- Build a draft progressive alignment
  - Determine pairwise similarity through k-mer counting (no alignment)
  - Construct tree
  - Construct draft progressive alignment following tree

MUSCLE

- Improve the progressive alignment
  - Compute pairwise similarity using the current MSA
  - Construct new tree with Kimura distance measure
  - Compare new and old trees: if improved, repeat this step, if not improved, then we're done

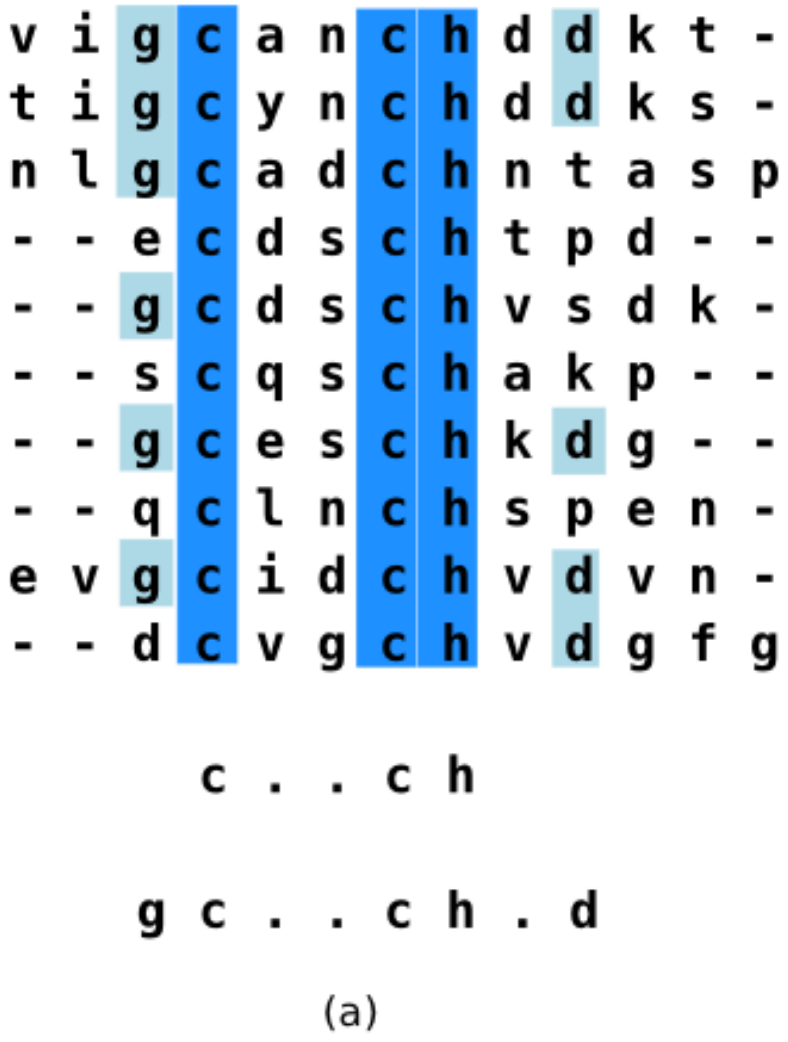
Multiple Alignment: Timeline

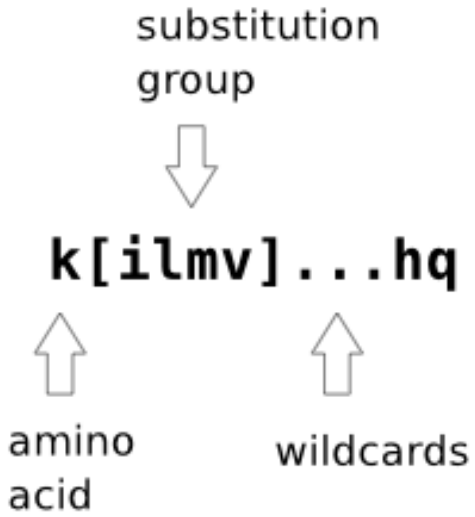
- **1975: Sankoff f**ormulated multiple alignment problem and gave dynamic programming solution
- **1990 Feng-Doolittle proposed P**rogressive alignment
- **1994 Thompson-Higgins-Gibson: ClustalW**
- Most popular multiple alignment program
- **2000 Notredame-Higgins-Heringa: T-coffee**
- Using the library of pairwise alignments
- **2004 MUSCLE**
- **2005 PROBCONS, MAFFT**

What can we do with an MSA?

- Represent it as a profile and use it for searching (alternative: HMM): PSI-BLAST
- Extract motifs: can then search for a particular feature of a protein

From MSA to discrete sequence motifs





Problems with the Formulation of MSA

- Multidomain proteins evolve not only through point mutations but also through domain duplications and domain recombinations
- Often impossible to align all protein sequences throughout their entire length
- Although MSA is a 30 year old problem, there were no MSA approaches for aligning **rearranged** sequences (i.e., multi-domain proteins with shuffled domains) prior to 2002

Alignment as a Graph

A

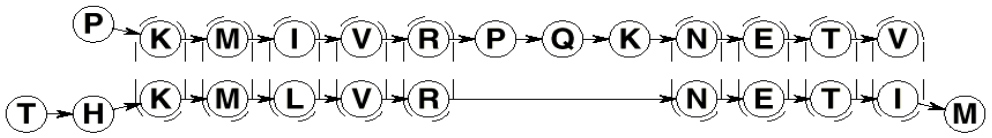
```
. . P K M I V R P Q K N E T V .
T H . K M L V R . . . N E T I M
```

Conventional Alignment

B

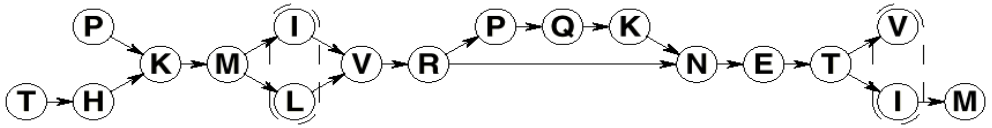
Sequence as a path

C



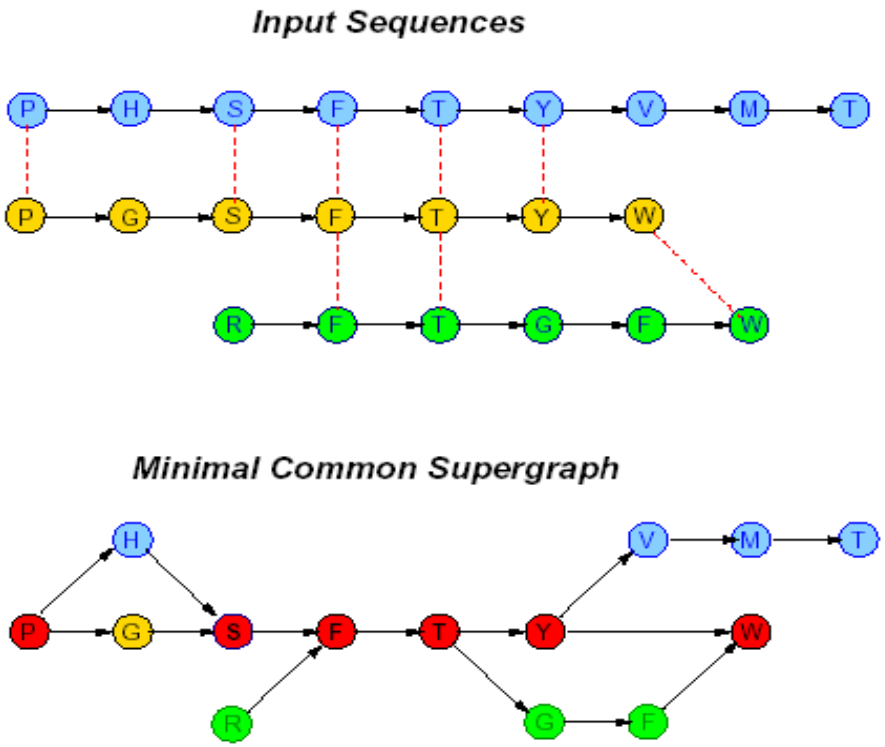
Two paths

D



Combined graph (partial order) of both sequences

Representing Sequences as Paths in a Graph



Each protein sequence is represented by a path. Dashed edges connect “equivalent” positions; vertices with identical labels are fused.

Partial Order Alignment (POA) Algorithm

Aligns sequences onto a directed acyclic graph (DAG)

Steps:

1. Guide Tree Construction
2. Progressive Alignment Following Guide Tree
3. Dynamic Programming Algorithm to align two PO-MSAs (PO-PO Alignment).

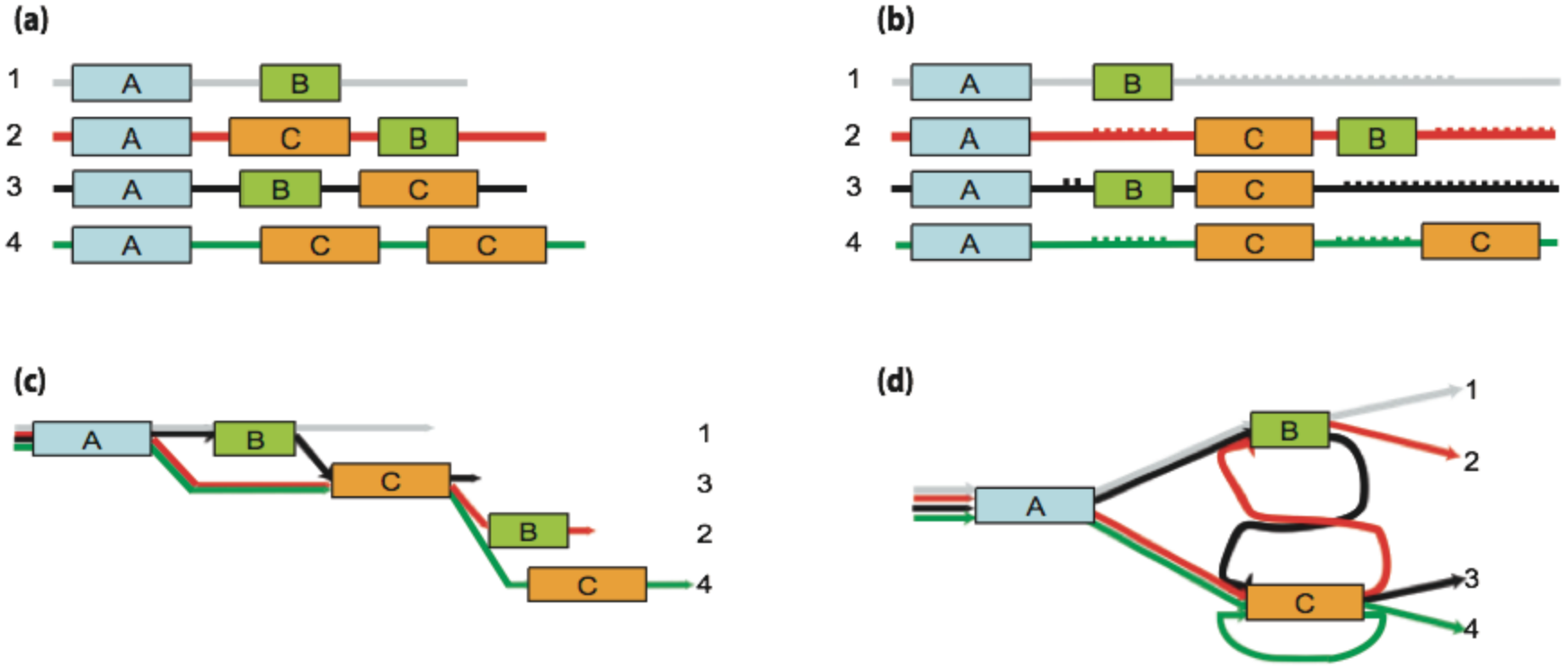
POA Advantages

- POA is more flexible: standard methods force sequences to align linearly
- PO-MSA representation handles gaps more naturally and retains (and uses) all information in the MSA

A-Bruijn Alignment (ABA)

- POA: represents alignment as directed graph; no cycles
- ABA: represents alignment as directed graph that may contains cycles

ABA vs. POA vs. MSA



Protein MSA programs

http://www.ebi.ac.uk/clustalw/

CLUSTALW – most widely used

http://phylogenomics.berkeley.edu/cgi-bin/muscle/input_muscle.py

MUSCLE – very scalable

http://mafft.cbrc.jp/alignment/software/

MAFFT – very scalable

http://probcons.stanford.edu/

PROBCONS – very accurate

http://www.bioinformatics.ucla.edu/poa/

POA - a different approach (very fast!)

http://tcoffee.org

T-Coffee – accurate, can incorporate other information (3D-Coffee)

Pattern advantages and disadvantages

Advantages:

- Relatively straightforward to identify (exact pattern matching is fast)
- Patterns are intuitive to read and understand
- Databases with large numbers of protein (e.g., PROSITE) and DNA sequence (e.g., JASPER and TRANSFAC) patterns are available.

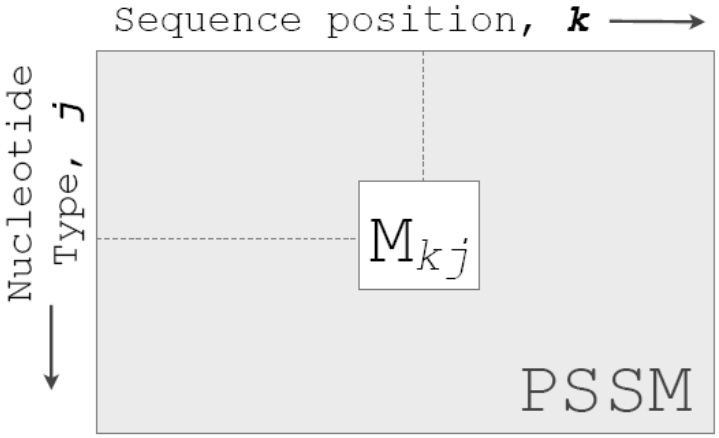
Disadvantages:

- Patterns are qualitative and *deterministic*
  (i.e., either matching or not!)
- We lose information about relative frequency of each residue at a position
  E.g., [GAC] *vs* 0.6 G, 0.28 A, and 0.12 C
- Can be difficult to write complex motifs using regular expression notation
- Cannot represent subtle sequence motifs

Profiles

- Motifs are qualitative pattern descriptors
  - No quantitative information
- Profiles
  - Describes a motif using quantitative information captured in a position specific scoring matrix (PSSM) (pronounced: POSSUM).
  - A simple PSSM has as many columns as there are positions in the alignment, and
  - either 4 rows (one for each DNA nucleotide) or 20 rows (one for each amino acid).
- You can search a database of sequences using a profile

PSSM



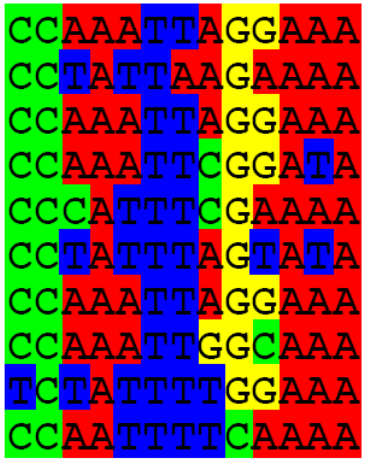
$$M_{kj} = \log\left(\frac{p_{kj}}{p_j}\right)$$

$\mathbf{M_{kj}}$ score for the *j*th nucleotide at position *k*

$\mathbf{p_{kj}}$ probability of nucleotide *j* at position *k*

$\mathbf{p_j}$ "background" probability of nucleotide *j*

Computing a transcription factor bind site PSSM



Alignment Counts Matrix:

Position k =	1	2	3	4	5	6	7	8	9	10	11	12	13
A:	0	0	6	10	5	0	1	5	0	3	10	8	10
C:	9	10	1	0	0	0	0	2	1	1	0	0	0
G:	0	0	0	0	0	0	0	1	9	5	0	0	0
T:	1	0	3	0	5	10	9	2	0	1	0	2	0
Consensus:	C	C	[ACT]	A	[AT]	T	T	N	G	N	A	[AT]	A

$$M_{kj} = \log\left(\frac{p_{kj}}{p_j}\right) \qquad p_{kj} = \frac{C_{kj} + p_j}{Z+1}$$

$$M_{kj} = \log\left(\frac{C_{kj} + p_j / Z + 1}{p_j}\right)$$

C_{kj} Number of *j*th type nucleotide at position *k*

Z Total number of aligned sequences

p_j "background" probability of nucleotide *j*

p_{kj} probability of nucleotide *j* at position *k*

Adapted from Hertz and Stormo, Bioinformatics 15:563-577

Computing a transcription factor bind site PSSM...

Alignment Matrix: C_{kj}

Position k =	1	2	3	4	5	6	7	8	9	10	11	12	13
A:	0	0	6	10	5	0	1	5	0	3	10	8	10
C:	9	10	1	0	0	0	0	2	1	1	0	0	0
G:	0	0	0	0	0	0	0	1	9	5	0	0	0
T:	1	0	3	0	5	10	9	2	0	1	0	2	0

k=1, j=A: $$M_{kj} = \log\left(\frac{C_{kj} + p_j / Z + 1}{p_j}\right) = \log\left(\frac{0 + 0.25/10 + 1}{0.25}\right) = -2.4$$

k=1, j=C: $$M_{kj} = \log\left(\frac{C_{kj} + p_j / Z + 1}{p_j}\right) = \log\left(\frac{9 + 0.25/10 + 1}{0.25}\right) = 1.2$$

k=1, j=T: $$M_{kj} = \log\left(\frac{C_{kj} + p_j / Z + 1}{p_j}\right) = \log\left(\frac{1 + 0.25/10 + 1}{0.25}\right) = -0.8$$

PSSM: M_{kj}

Position k =	1	2	3	4	5	6	7	8	9	10	11	12	13
A:	-2.4	-2.4	0.8	1.3	0.6	-2.4	-0.8	0.6	-2.4	0.2	1.3	1.1	1.3
C:	1.2	1.3	-0.8	-2.4	-2.4	-2.4	-2.4	-0.2	-0.8	-0.8	-2.4	-2.4	-2.4
G:	-2.4	-2.4	-2.4	-2.4	-2.4	-2.4	-2.4	-0.8	1.2	0.6	-2.4	-2.4	-2.4
T:	-0.8	-2.4	0.2	-2.4	0.6	1.3	1.2	-0.2	-2.4	-0.8	-2.4	-0.2	-2.4

Scoring a test sequence

Query Sequence

CCTATTTAGGATA

PSSM:

Position k =	1	2	3	4	5	6	7	8	9	10	11	12	13
A:	-2.4	-2.4	0.8	**1.3**	0.6	-2.4	-0.8	**0.6**	-2.4	0.2	**1.3**	1.1	**1.3**
C:	**1.2**	**1.3**	-0.8	-2.4	-2.4	-2.4	-2.4	-0.2	-0.8	-0.8	-2.4	-2.4	-2.4
G:	-2.4	-2.4	-2.4	-2.4	-2.4	-2.4	-2.4	-0.8	**1.2**	**0.6**	-2.4	-2.4	-2.4
T:	-0.8	-2.4	**0.2**	-2.4	**0.6**	**1.3**	**1.2**	-0.2	-2.4	-0.8	-2.4	**-0.2**	-2.4
Test seq:	C	C	T	A	T	T	T	A	G	G	A	T	A

Query Score = 1.2 + 1.3 + 0.2 + 1.3 + 0.6 + 1.3 + 1.2
+ 0.6 + 1.2 + 0.6 + 1.3 + −0.2 + 1.3
= 11.9

Q. Does the query sequence match the DNA sequence profile?

Scoring a test sequence...

Query Sequence

CCTATTTAGGATA

Best Possible Sequence

CCAATTTAGGAAA

PSSM:

Position k =	1	2	3	4	5	6	7	8	9	10	11	12	13
A:	-2.4	-2.4	**0.8**	**1.3**	0.6	-2.4	-0.8	**0.6**	-2.4	0.2	**1.3**	**1.1**	**1.3**
C:	**1.2**	**1.3**	-0.8	-2.4	-2.4	-2.4	-2.4	-0.2	-0.8	-0.8	-2.4	-2.4	-2.4
G:	-2.4	-2.4	-2.4	-2.4	-2.4	-2.4	-2.4	-0.8	**1.2**	**0.6**	-2.4	-2.4	-2.4
T:	-0.8	-2.4	0.2	-2.4	**0.6**	**1.3**	**1.2**	-0.2	-2.4	-0.8	-2.4	-0.2	-2.4
Max Score:	**C**	**C**	**A**	**A**	**T**	**T**	**T**	**A**	**G**	**G**	**A**	**A**	**A**

Max Score = 1.2 + 1.3 + 0.8 + 1.3 + 0.6 + 1.3 + 1.2
+ 0.6 + 1.2 + 0.6 + 1.3 + 1.1 + 1.3
= 13.8

A. Following method in Harbison *et al.* (2004) Nature 431:99-104

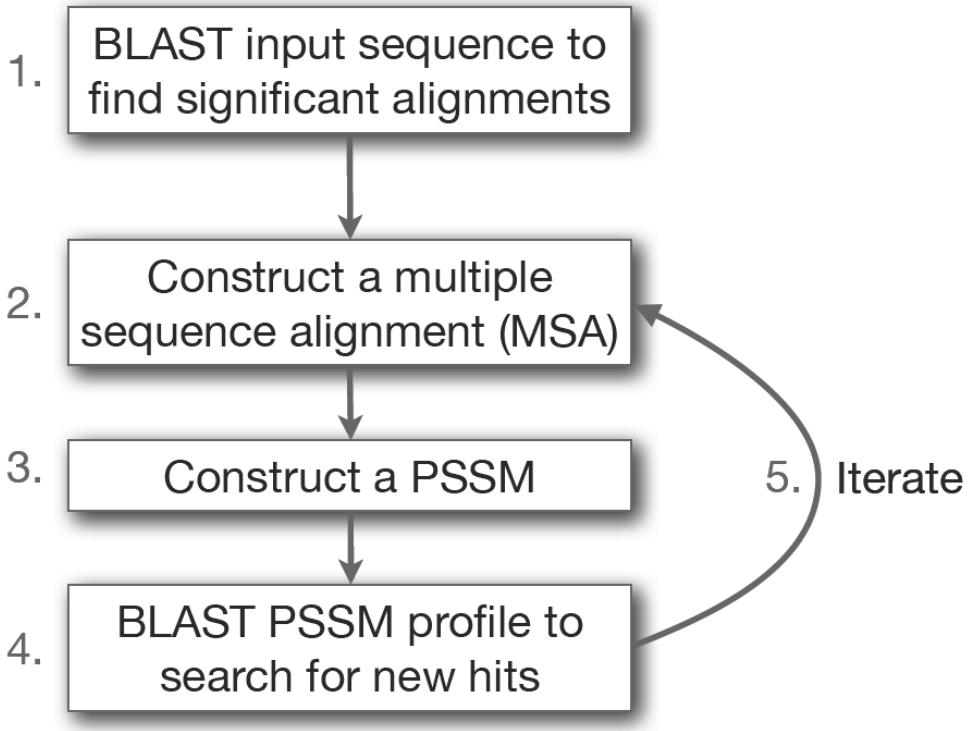
Heuristic threshold for match = 60% x Max Score = (0.6 x 13.8 = 8.28);
11.9 > 8.28; Therefore our query is a potential TFBS!

Detecting Remote Homology

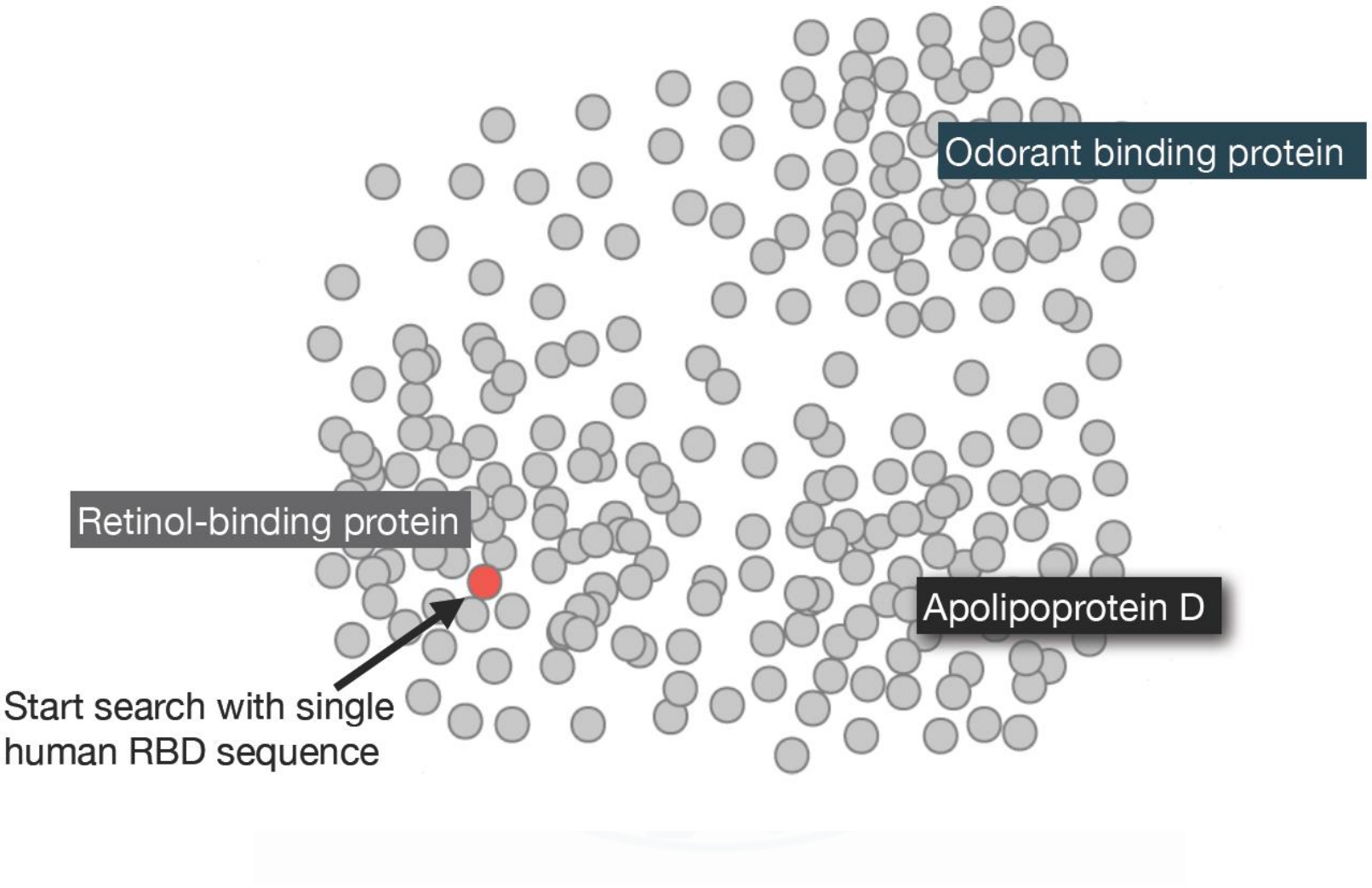
- In order to find distant homologs of proteins
  - PSI-BLAST
  - HMMER
  - HHBLITZ

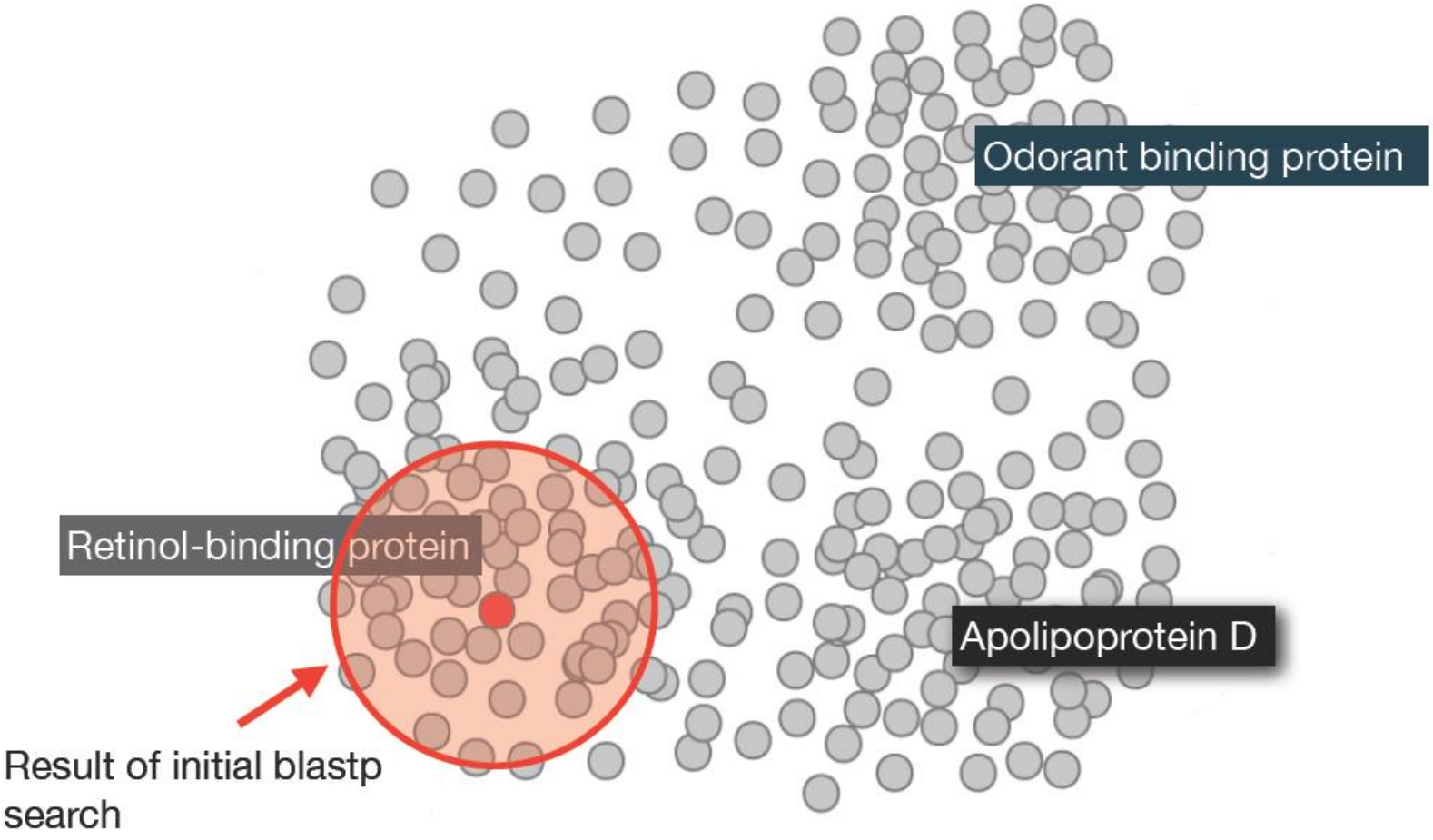
PSI-BLAST: Position-Specific Iterated BLAST

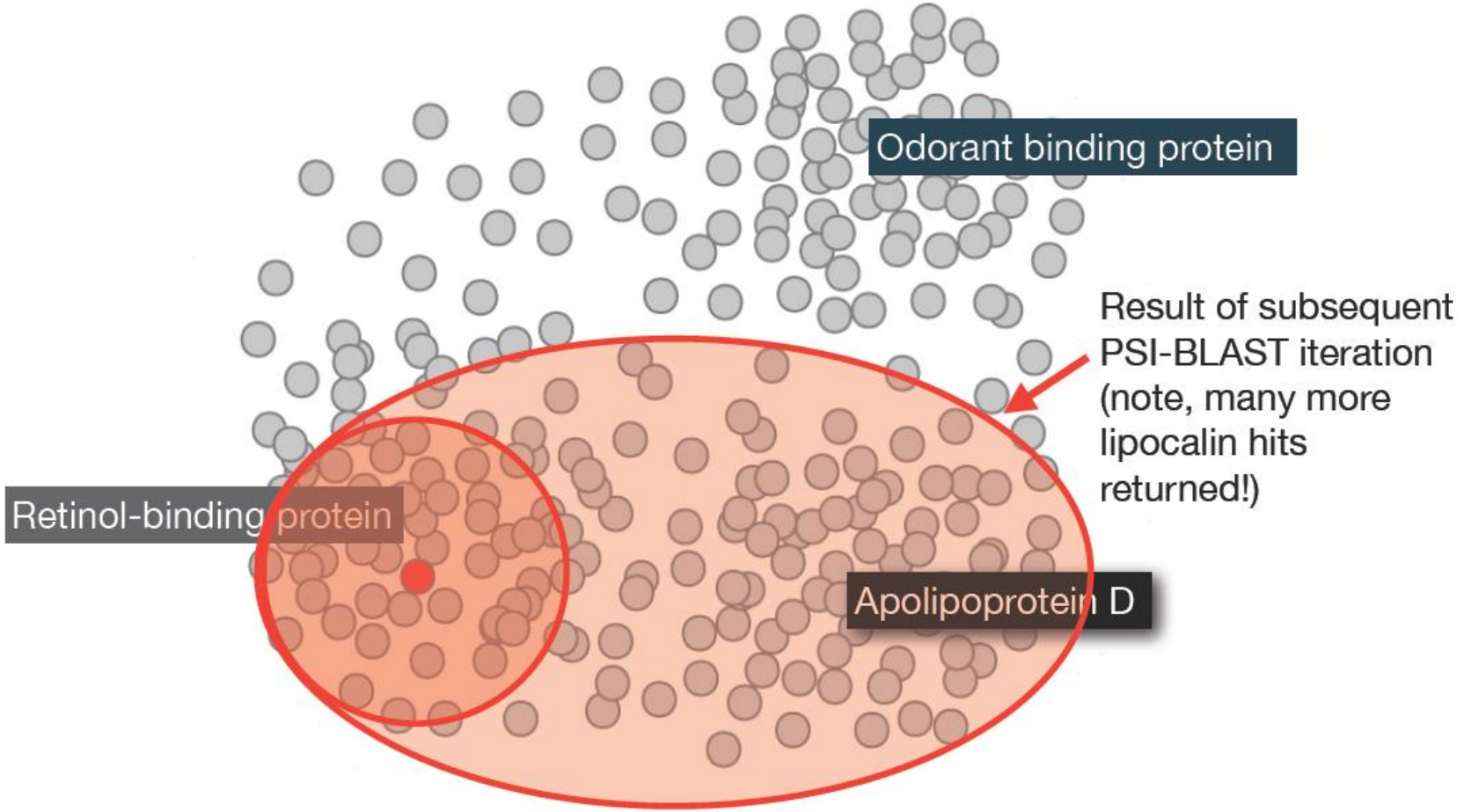
Many proteins in a database are too distantly related to a query to be detected using standard BLAST. In many other cases matches are detected but are so distant that the inference of homology is unclear. Enter the more sensitive PSI-BLAST



(see Altschul *et al.*, Nuc. Acids Res. (1997) 25:3389-3402)


Odorant binding protein
Retinol-binding protein
Apolipoprotein D
Start search with single
human RBD sequence


Odorant binding protein
Retinol-binding protein
Apolipoprotein D
Result of initial blastp search


Odorant binding protein
Result of subsequent PSI-BLAST iteration (note, many more lipocalin hits returned!)
Retinol-binding protein
Apolipoprotein D

PSI-BLAST returns dramatically more hits

PSI-BLAST frequently returns many more hits with significant E-values than blastp

The search process is continued iteratively, typically about five times, and at each step a new PSSM is built.

- You must decide how many iterations to perform and which sequences to include!
  You can stop the search process at any point - typically whenever few new results are returned or when no new "sensible" results are found.

Iteration	Hits with E < 0.005	Hits with E > 0.005
1	34	61
2	314	79
3	416	57
4	432	50
5	432	50

Human retinol-binding protein 4 (RBP4; P02753) was used as a query in a PSI-BLAST search of the RefSeq database.

(**a**) Iteration 1

```
>ref|NP_001638.1| apolipoprotein D precursor [Homo sapiens]
Length=189

 Score = 57.4 bits (137),  Expect = 3e-07, Method: Composition-based stats.
 Identities = 47/151 (31%), Positives = 78/151 (51%), Gaps = 39/151 (25%)

Query  29   VKENFDKARFSGTWYAMAKKDPEGLFLQDNIVAEFSVDETGQMSATAKGRVRLLNNWDVC  88
            V+ENFD  ++ G WY + +K P         I A +S+ E G        ++++LN  ++
Sbjct  33   VQENFDVNKYLGRWYEI-EKIPTTFENGRCIQANYSLMENG--------KIKVLNQ-ELR  82

Query  89   ADMVGTFTDTE---------DPAKFKMKY-WGVASFLQKGNDDHWIVDTDYDTYAVQYSC  138
            AD  GT    E         +PAK ++K+ W + S        +WI+ TDY+ YA+ YSC
Sbjct  83   AD--GTVNQIEGEATPVNLTEPAKLEVKFSWFMPS------APYWILATDYENYALVYSC  134

Query  139  ----RLLNLDGTCADSYSFVFSRDPNGLPPE  165
                +L ++D      ++++ +R+PN LPPE
Sbjct  135  TCIIQLFHVD------FAWILARNPN-LPPE  158
```

(**b**) Iteration 2

```
>ref|NP_001638.1| apolipoprotein D precursor [Homo sapiens]
Length=189

 Score =  175 bits (443),  Expect = 1e-42, Method: Composition-based stats.
 Identities = 45/163 (27%), Positives = 77/163 (47%), Gaps = 31/163 (19%)

Query  14   GSGRAERDCRVSSFRVKENFDKARFSGTWYAMAKKDPEGLFLQDNIVAEFSVDETGQMSA  73
              G+A    +  +  V+ENFD  ++ G WY + +K P         I A +S+ E G++
Sbjct  18   AEGQAFHLGKCPNPPVQENFDVNKYLGRWYEI-EKIPTTFENGRCIQANYSLMENGKIKV  76

Query  74   TAK-----GRVRLLNNWDVCADMVGTFTDTEDPAKFKMKY-WGVASFLQKGNDDHWIVDT  127
              +     G V  +           T  +  +PAK ++K+ W + S        +WI+ T
Sbjct  77   LNQELRADGTVNQIEG-------EATPVNLTEPAKLEVKFSWFMPS------APYWILAT  123

Query  128  DYDTYAVQYSCR----LLNLDGTCADSYSFVFSRDPNGLPPEA  166
            DY+ YA+ YSC     L ++D      ++++ +R+PN LPPE
Sbjct  124  DYENYALVYSCTCIIQLFHVD------FAWILARNPN-LPPET  159
```

(**c**) Iteration 3

```
>ref|NP_000597.1| complement component 8, gamma polypeptide [Homo sapiens]
Length=202

 Score =  104 bits (260),  Expect = 2e-21, Method: Composition-based stats.
 Identities = 40/186 (21%), Positives = 74/186 (39%), Gaps = 29/186 (15%)

Query  24   VSSFRVKENFDKARFSGTWYAMAKKDPEGLFLQDNIVAEFSVDETG-QMSATAKGRVRLL  82
            +S+ + K NFD  +F+GTW  +A          +    AE +     Q +A A    R L
Sbjct  33   ISTIQPKANFDAQQFAGTWLLVAVGSACRFLQEQGHRAEATTLHVAPQGTAMAVSTFRKL  92

Query  83   NNWDVCADMVGTFTDTEDPAKFKMKYWGVASFLQKGNDDHWIVDTDYDTYAVQY------  136
            +   +C  +   + DT    +F ++  G      +G     + +TDY ++AV Y
Sbjct  93   DG--ICWQVRQLYGDTGVLGRFLLQARGA-----RGAVHVVVAETDYQSFAVLYLERAGQ  145

Query  137  -SCRLLNLDGTCADSYSFVFSRDPNGLPPEAQKIVRQRQEELCLARQYRLIVHNGYCDGR  195
             S +L       +DS    F +       EA       ++++    +Y      G+C+
Sbjct  146  LSVKLYARSLPVSDSVLSGFEQRVQ----EA----HLTEDQIFYFPKY------GFCEAA  191

Query  196  SERNLL  201
             + ++L
Sbjct  192  DQFHVL  197
```

blastp E-value for this hit was 0.27

Example PSI-BLAST PSSM at iteration 3

The PSI-BLAST PSSM is essentially a query customized scoring matrix that is more sensitive than PAM or BLOSUM (e.g. BLOSUM S_{AA} = +4)

20 amino acids types

Query residues/positions

		A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y	V
1	M	-1	-2	-2	-3	-2	-1	-2	-3	-2	1	2	-2	6	0	-3	-2	-1	-2	-1	1
2	K	-1	1	0	1	-4	2	4	-2	0	-3	-3	3	-2	-4	-1	0	-1	-3	-2	-3
3	W	-3	-3	-4	-5	-3	-2	-3	-3	-3	-3	-2	-3	-2	1	-4	-3	-3	12	2	-3
4	V	0	-3	-3	-4	-1	-3	-3	-4	-4	3	1	-3	1	-1	-3	-2	0	-3	-1	4
5	W	-3	-3	-4	-5	-3	-2	-3	-3	-3	-3	-2	-3	-2	1	-4	-3	-3	12	2	-3
6	A	5	-2	-2	-2	-1	-1	-1	0	-2	-2	-2	-1	-1	-3	-1	1	0	-3	-2	0
7	L	-2	-2	-4	-4	-1	-2	-3	-4	-3	2	4	-3	2	0	-3	-3	-1	-2	-1	1
8	L	-1	-3	-3	-4	-1	-3	-3	-4	-3	2	2	-3	1	3	-3	-2	-1	-2	0	3
9	L	-1	-3	-4	-4	-1	-2	-3	-4	-3	2	4	-3	2	0	-3	-3	-1	-2	-1	2
10	L	-2	-2	-4	-4	-1	-2	-3	-4	-3	2	4	-3	2	0	-3	-3	-1	-2	-1	1
11	A	5	-2	-2	-2	-1	-1	-1	0	-2	-2	-2	-1	-1	-3	-1	1	0	-3	-2	0
12	A	5	-2	-2	-2	-1	-1	-1	0	-2	-2	-2	-1	-1	-3	-1	1	0	-3	-2	0
13	W	-2	-3	-4	-4	-2	-2	-3	-4	-3	1	4	-3	2	1	-3	-3	-2	7	0	0
14	A	3	-2	-1	-2	-1	-1	-2	4	-2	-2	-2	-1	-2	-3	-1	1	-1	-3	-3	-1
15	A	2	-1	0	-1	-2	2	0	2	-1	-3	-3	0	-2	-3	-1	3	0	-3	-2	-2
16	A	4	-2	-1	-2	-1	-1	-1	3	-2	-2	-2	-1	-1	-3	-1	1	0	-3	-2	-1
...																					
37	S	2	-1	0	-1	-1	0	0	0	-1	-2	-3	0	-2	-3	-1	4	1	-3	-2	-2
38	G	0	-3	-1	-2	-3	-2	-2	6	-2	-4	-4	-2	-3	-4	-2	0	-2	-3	-3	-4
39	T	0	-1	0	-1	-1	-1	-1	-2	-2	-1	-1	-1	-1	-2	-1	1	5	-3	-2	0
40	W	-3	-3	-4	-5	-3	-2	-3	-3	-3	-3	-2	-3	-2	1	-4	-3	-3	12	2	-3
41	Y	-2	-2	-2	-3	-3	-2	-2	-3	2	-2	-1	-2	-1	3	-3	-2	-2	2	7	-1
42	A	4	-2	-2	-2	-1	-1	-1	0	-2	-2	-2	-1	-1	-3	-1	1	0	-3	-2	0
...																					

PSI-BLAST errors: the corruption problem

The main source of error in PSI-BLAST searches is the spurious amplification of sequences that are unrelated to the query.

There are three main approaches to stopping corruption of PSI-BLAST queries:

- Perform multi-domain splitting of your query sequence
  If a query protein has several different domains PSI-BLAST may find database matches related to both individually. One should not conclude that these hits with different domains are related.
  - Often best to search using just one domain of interest.

- Inspect each PSI-BLAST iteration removing suspicious hits.
  E.g., your query protein may have a generic coiled-coil domain, and this may cause other proteins sharing this motif (such as myosin) to score better than the inclusion threshold even though they are not related.
  - Use your biological knowledge!

- Lower the default expect level (e.g., E = 0.005 to E = 0.0001).
  This may suppress appearance of FPs (but also TPs)

Profile advantages and disadvantages

Advantages:

- Quantitate with a good scoring system
- Weights sequences according to observed diversity
  Profile is specific to input sequence set
- Very sensitive
  Can detect weak similarity
- Relatively easy to compute
  Automatic profile building tools available

Disadvantages:

- If a mistake enters the profile, you may end up with irrelevant data
  The corruption problem!
- Ignores higher order dependencies between positions
  i.e., correlations between the residue found at a given position and those found at other positions (e.g. salt-bridges, structural constraints on RNA etc...)
- Requires some expertise to use proficiently

HMMER

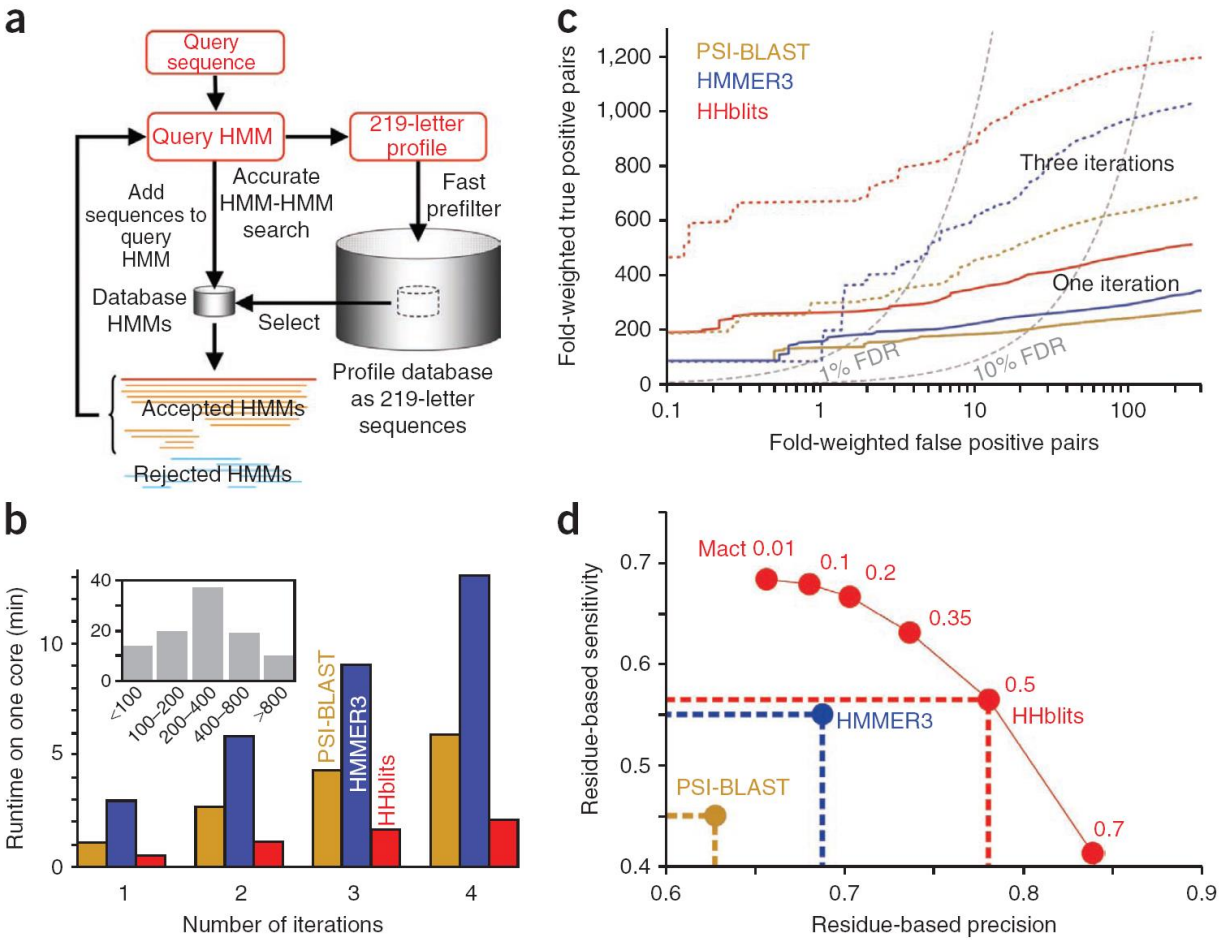
- Builds a profile Hidden Markov Model (HMMR) and compares a query sequence to it
- Profile-HMMs are constructed from Multiple Sequence Alignment using hmmbuild
- Used in protein family databases and software
  - Pfam
  - InterPro
  - UGENE
- HMMER3
  - Latest version
  - Speed comparable to BLAST

InterPro

- Database of protein families, domains and functional sites in which identifiable features found in known proteins can be applied to new protein sequences in order to functionally characterize them
  - Can search the database using regular expressions or Hidden Markov Models
- Domains are standalone functional units in proteins
- Proteins having the same domain are said to belong to the same family

HHBlitz: Remmert (2012)

Figure 1 | Workflow and benchmark comparison. (**a**) HHblits can iteratively search for homologous sequences in large databases such as UniProt. The HHblits database is a clustered version in which each set of full-length alignable sequences is represented by an HMM. Sequences from matched HMMs with a statistically significant *E* value are added to the query MSA, from which a new HMM is calculated for the next search iteration. A prefilter reduces the number of full HMM-HMM alignments by ~2,500-fold. (**b**) Median run times for searches with 100 test sequences through the UniProt or UniProt20 database (the inset shows the test sequence length distribution). (**c**) True positive pairs (same SCOP fold) compared to false positive pairs (different SCOP fold) for one and three search iterations in an all-against-all comparison. FDR, false discovery rate. (**d**) Mean fraction of correctly aligned residue pairs out of all structurally alignable pairs (sensitivity) compared to the fraction of correctly aligned pairs out of all the aligned pairs (precision). The parameter mact controls the alignment greediness (**Supplementary Fig. 10**).



HHBlitz application

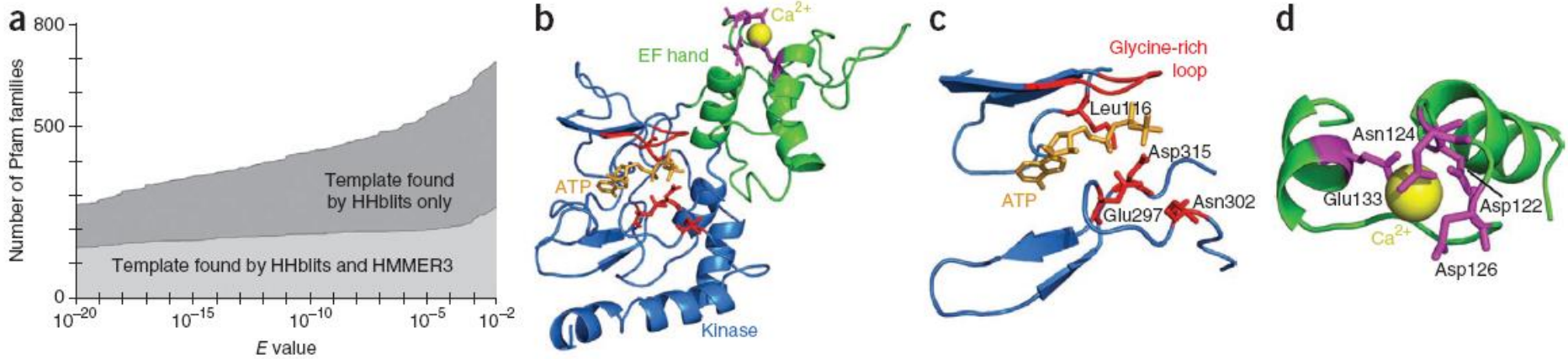


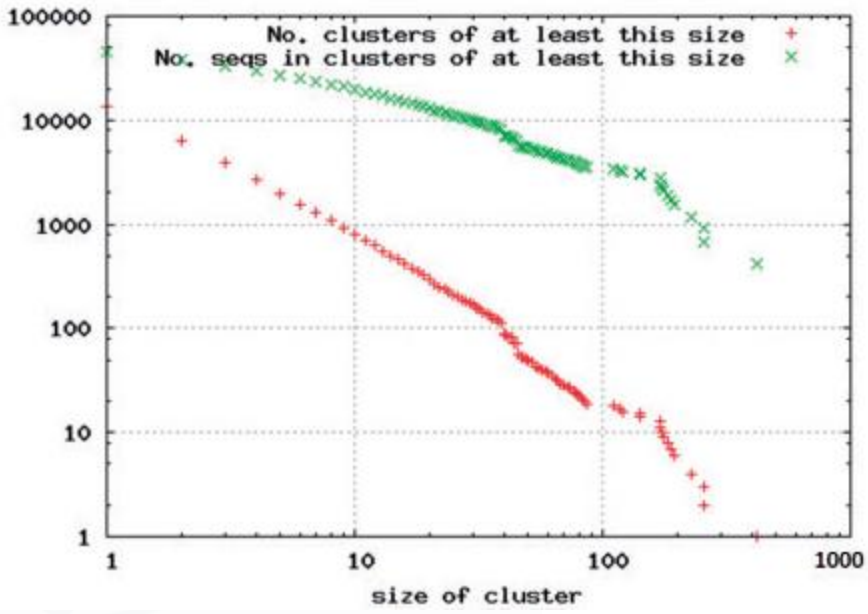
Figure 2 | Structure predictions for Pfam families and the modeling of human Pip49 (also known as FAM69B). (**a**) Families to which only HHblits and both HHblits and HMMER3 assigned a structural template below a given *E* value. (**b**) Homology model of human Pip49 kinase domain (blue) with the inserted EF hand (green). (**c**) Catalytic center showing the conserved residues (red) for protein kinase activity. (**d**) EF hand insertion with the conserved residues (magenta) for the predicted Ca^{2+}-dependent activation.

Remote homology detection using Machine Learning

- "Remote homology detection: a motif based approach" Ben-Hur and Brutlag, Bioinformatics (2003)

Sequence clustering

- Create non-redudant data sets
- BLASTCLUST
- CD-HIT
  - Huang (2010)
  - Sequences with more than T% maximum similarity over a W lengtl window are clustered into the same cluster



InterPro

- Contains
  - Pfam: large collection of MSAs and HMMs covering any common protein domains and families
  - PROSITE : database of protein families and domains – it consists of biologically significant sites, patterns and profiles that help to reliably identidfy to which known protein family (if any) a new sequence belongs
  - ProDom, SMART, SUPERFamily….

How is this different from conventional database search?

- You want to find out
  - Exact and inexact matches
  - Express sequences differently

Protein Search	Profile Search (Iterative)
BLAST	PSI-BLAST
HHSearch	HHBlits

Using Profiles

- There are numerous uses of profiles
  - As features in prediction algorithms
    - PSI-BLAST/HHBlits PSSM is used in PAIRpred for protein interface prediction. Aligned against Non-redundant database.
    - Can use HHBlits profile for CAM binding prediction

End of Lecture