Lectures 20, 21: Multiple Sequence Alignment

Fall 2018
Nov 27, 29, 2018
Multiple Alignment vs Pairwise Alignment

- Up until now we have only tried to align two sequences.
Multiple Alignment vs Pairwise Alignment

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

- A faint similarity between two sequences becomes significant if present in many
- Multiple alignments can reveal subtle similarities that pairwise alignments do not reveal

“Pairwise alignment whispers... multiple alignment shouts out loud” Hubbard, Lesk, Tramontano, Nature Structural Biology 1996.
MSA: why?

- Identify regions in the protein that have an important role in the structure and function of a group of related proteins: conservation implies importance
- Construct *motifs/domains* that describe those regions
MSA: why?

- Identify and represent protein families
- Deduce evolutionary history
Genome multiple alignment?

- Yes, but need special methods for that
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

\[
\begin{align*}
\text{A} & \quad \text{T} & \quad - & \quad \text{G} & \quad \text{C} & \quad \text{G} & \quad - \\
\text{A} & \quad - & \quad \text{C} & \quad \text{G} & \quad \text{T} & \quad - & \quad \text{A} \\
\text{A} & \quad \text{T} & \quad \text{C} & \quad \text{A} & \quad \text{C} & \quad - & \quad \text{A}
\end{align*}
\]

- Score: more conserved columns, better alignment
Alignment = Paths in...

Align 3 sequences: ATGC, AATC, ATGC

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>--</th>
<th>T</th>
<th>G</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>A</td>
<td>T</td>
<td></td>
<td></td>
<td>C</td>
</tr>
<tr>
<td></td>
<td>--</td>
<td>A</td>
<td>T</td>
<td>G</td>
<td>C</td>
</tr>
</tbody>
</table>
Alignment Paths

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>1</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>--</td>
<td>T</td>
<td>G</td>
<td>C</td>
<td></td>
<td>x coordinate</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>A</th>
<th>A</th>
<th>T</th>
<th>--</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>--</td>
<td>A</td>
<td>T</td>
<td>G</td>
<td>C</td>
</tr>
</tbody>
</table>
### Alignment Paths

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>--</td>
<td>T</td>
<td>G</td>
<td>C</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>A</td>
<td>T</td>
<td>--</td>
<td>C</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>--</th>
<th>A</th>
<th>T</th>
<th>G</th>
<th>C</th>
</tr>
</thead>
</table>

`x coordinate`

`y coordinate`
Alignment Paths

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>1</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>x coordinate</td>
<td>A</td>
<td>--</td>
<td>T</td>
<td>G</td>
<td>C</td>
<td></td>
</tr>
<tr>
<td>y coordinate</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>z coordinate</td>
<td>A</td>
<td>A</td>
<td>T</td>
<td>--</td>
<td>C</td>
<td></td>
</tr>
</tbody>
</table>

- 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

\[ s_{i,j,k} = \max \left\{ \begin{array}{l}
  s_{i-1,j-1,k-1} + \delta(v_i, w_j, u_k) \\
  s_{i-1,j-1,k} + \delta(v_i, w_j, -) \\
  s_{i-1,j,k-1} + \delta(v_i, -, u_k) \\
  s_{i,j-1,k-1} + \delta(-, w_j, u_k) \\
  s_{i-1,j,k} + \delta(v_i, -, -) \\
  s_{i,j-1,k} + \delta(-, w_j, -) \\
  s_{i,j,k-1} + \delta(-, -, u_k) \\
\end{array} \right. \]

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.
Additive MSA scores

• General structure of an MSA scoring scheme:

\[ S(m) = G + \sum_i S(m_i) \]

• Where \( m \) is the alignment, \( m_i \) is column \( i \) of the alignment and \( S(m_i) \) is the score of column \( i \), and \( G \) scores the gaps.

• Assumption: columns are independent
Sum of Pairs (SP) score

\[ S(m_i) = \sum_{j<k} \delta(m_{ij}, m_{ik}) \]

ATG–C–AAT
A–G–CATAT
ATCCCATTT

Score=3

Score = 1 – 2\mu
Problem

- MSA with SP-score is NP-complete
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 0 1 0 0 0 .8 0 0 0 0 0
C .6 0 0 0 0 1 0 0 0 .4 1 0 .6 .2 0 0 0
G 0 0 1 .2 0 0 0 0 0 0 .2 0 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 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
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  -

sequences

distance matrix

guide tree (neighbor-joining)

progressive alignment following guide tree

1  PEEKSAVTALWGKVN--VDEVGG
2  GEEKAAVLALWDKVN--EEEVGG
3  PADKTNVKAAGWGVGAHAGEYGA
4  AADKTNVKAASKWGVGGHAGEYGA
5  EHEWQLVLHVWAKVEADVAGHGQ
Progressive alignment - step 1

gctcgataacgatacagatgactagcta

gctcgataacaagacgatgacagcta

gctcgataacagatgactagcta

gctcgataacagatgacgagcga

tctgaacgatacagactagct

---

gctcgataacgatacagatgactagcta

gctcgataacaagacgatgac-gacta

gctcgataacagatgactagcta

gctcgataacaagacgagctagct

gctcgataacagatgacgagcga

tctgaacgatacagactagct
Progressive alignment - step 2

gctcgatacagatgatacgatacgatgactagcta
gctcgatacacaagacgatgacagacta
gctcgatacagatgactagcta
gctcgatacagatgacgagcga
tcgaacgatacgatgactagct

gctcgatacagatgactagcta
gctcgatacacaagacgatgacgagcga
Progressive alignment - step 3

gctcgatacgcgatagctagctagctagcta
gctgcatacaagacgatgac-agctagcta
+
gctcgatacagctagctagctagctagcta

gctcgatacagctagctagctagcta

gctcgatacagctagctagctagcta

gctcgatacagctagctagctagcta

---
tgactagcta
---
tgactagcta

gctcgatacagctagctagctagcta

gctcgatacagctagctagctagcta

gctcgatacagctagctagctagcta

gctcgatacagctagctagctagcta

---
tgactagcta
---
tgactagcta

gctcgatacagctagctagctagcta

gctcgatacagctagctagctagcta

---
tgactagcta
---
tgactagcta

gctcgatacagctagctagctagcta

gctcgatacagctagctagctagcta

---
tgactagcta
---
tgactagcta

gctcgatacagctagctagctagcta

gctcgatacagctagctagctagcta

---
tgactagcta
---
tgactagcta

Progressive alignment - final step

gctcgatacgcgatacgcgatgactagctica

gctcgatacaacaagacgatgac-agcta

gctcgatacaca-cga---tgactagctica

gctcgatacacaca-cga---tgacgacgca

+ 

gctcgatacacacgactagctica

gctcgatacacaca-cga---tgacgacgca

---

tctgaacgatacgcgatgactagctica

gctcgatacacaca-cga---tgacgacgca

gctcgatacacacaca-cga---tgacgacgca

---

tctga-acgatacgcgatacgactagctica
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...

Sequence A: GARFIELD THE LAST FAT CAT
Sequence B: GARFIELD THE FAST CA-T
Sequence C: GARFIELD THE VERY FAST CAT
Sequence D: 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:

\[
\begin{align*}
  x &: \text{GAAGTT} \\
  y &: \text{GAC–TT} \\
  z &: \text{GAACTG} \\
  w &: \text{GTACTG}
\end{align*}
\]

Frozen!

Now clear that \( y = \text{GA- CTT} \) 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 are done
Multiple Alignment: Timeline

1975 Sankoff
   Formulated multiple alignment problem and gave dynamic programming solution

1990 Feng-Doolittle
   Progressive alignment

1994 Thompson-Higgins-Gibson: ClustalW
   Most popular multiple alignment program

   Using the library of pairwise alignments

2004 MUSCLE

2005 PROBCONS, MAFFT

2012 Alignathon Contests
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


Conventional Alignment

B \begin{align*}
P &
\rightarrow K \\
\rightarrow M \\
\rightarrow I \\
\rightarrow V \\
\rightarrow R \\
\rightarrow P \\
\rightarrow Q \\
\rightarrow K \\
\rightarrow N \\
\rightarrow E \\
\rightarrow T \\
\rightarrow V \\
\end{align*}

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 contain 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)