Survey of Clustal family of alignment methods

Rishika Agarwal
Clustal methods

1. Clustal
2. Clustal V
3. Clustal W
4. Clustal X
5. MultiClustal
6. DbClustal
7. Clustal Omega
Clustal - 1988
Progressive alignment method
Clustal - 1988

Progressive alignment method

• Pairwise distance matrix - Wilbur and Lipman’s (WL) algorithm
Clustal - 1988

Progressive alignment method

- Pairwise distance matrix - Wilbur and Lipman’s (WL) algorithm
- Dendrogram construction - UPGMA
Clustal - 1988

Progressive alignment method

• Pairwise distance matrix - Wilbur and Lipman’s (WL) algorithm
• Dendrogram construction - UPGMA
• Progressive alignment - WL algorithm
Clustal - 1988

Progressive alignment method

• Pairwise distance matrix - Wilbur and Lipman’s (WL) algorithm
• Dendrogram construction - UPGMA
• Progressive alignment - WL algorithm

• Consensus sequences used to represent cluster of sequences below them
Clustal - 1988

Progressive alignment method

- Pairwise distance matrix - Wilbur and Lipman’s (WL) algorithm
- Dendrogram construction - UPGMA
- Progressive alignment - WL algorithm

- Consensus sequences used to represent cluster of sequences below them
- Fixed Substitution Weight Matrix fixed for residues - Dayhoff matrix
Clustal - 1988

Progressive alignment method

• Pairwise distance matrix - Wilbur and Lipman’s (WL) algorithm
• Dendrogram construction - UPGMA
• Progressive alignment - WL algorithm

• Consensus sequences used to represent cluster of sequences below them
• Fixed Substitution Weight Matrix fixed for residues - Dayhoff matrix

System details:
Clustal - 1988

Progressive alignment method

• Pairwise distance matrix - Wilbur and Lipman’s (WL) algorithm
• Dendrogram construction - UPGMA
• Progressive alignment - WL algorithm

• Consensus sequences used to represent cluster of sequences below them
• Fixed Substitution Weight Matrix fixed for residues - Dayhoff matrix

System details :
• Language : Fortran
Clustal - 1988

Progressive alignment method

- Pairwise distance matrix - Wilbur and Lipman’s (WL) algorithm
- Dendrogram construction - UPGMA
- Progressive alignment - WL algorithm
- Consensus sequences used to represent cluster of sequences below them
- Fixed Substitution Weight Matrix fixed for residues - Dayhoff matrix

System details:
- Language: Fortran
- Hardware requirements: microcomputer
Clustal V - 1992
Clustal V - 1992

Major differences:
Major differences:

- Flexible input and output formats. Could read gaps (-) in input sequences
Clustal V - 1992

Major differences:

- Flexible input and output formats. Could read gaps (-) in input sequences
- Progressive alignment: aligns two alignments instead of pairwise sequence alignments with consensus sequence
Clustal V - 1992

Major differences:

• Flexible input and output formats. Could read gaps (-) in input sequences

• Progressive alignment: aligns two alignments instead of pairwise sequence alignments with consensus sequence

• Allows construction of refined phylogenetic tree after the MSA - Neighbor Joining
Clustal W - 1994

• Two major problems of Clustal:
Clustal W - 1994

- Two major problems of Clustal:
  1. Local minimum problem:
Two major problems of Clustal:

1. **Local minimum problem**:
   - Greedy algorithm may get stuck in local minima, and mistakes get accumulated at each step.
• Two major problems of Clustal:

1. **Local minimum problem**:
   • Greedy algorithm may get stuck in local minima, and mistakes get accumulated at each step.
   • Need iterative/stochastic procedures to solve this
Two major problems of Clustal:

1. **Local minimum problem**:
   - Greedy algorithm may get stuck in local minima, and mistakes get accumulated at each step.
   - Need iterative/stochastic procedures to solve this

2. **Choice of alignment parameters**:
Two major problems of Clustal:

1. **Local minimum problem:**
   - Greedy algorithm may get stuck in local minima, and mistakes get accumulated at each step.
   - Need iterative/stochastic procedures to solve this.

2. **Choice of alignment parameters:**
   - Free parameters - weight matrix, gap opening and gap extension penalties.
Two major problems of Clustal:

1. **Local minimum problem**:
   - Greedy algorithm may get stuck in local minima, and mistakes get accumulated at each step.
   - Need iterative/stochastic procedures to solve this

2. **Choice of alignment parameters**:
   - Free parameters - weight matrix, gap opening and gap extension penalties.
   - For similar sequences, many possible values
• Two major problems of Clustal:

1. **Local minimum problem:**
   • Greedy algorithm may get stuck in local minima, and mistakes get accumulated at each step.
   • Need iterative/stochastic procedures to solve this

2. **Choice of alignment parameters:**
   • Free parameters - weight matrix, gap opening and gap extension penalties.
   • For similar sequences, many possible values
   • Different weight matrices and gap penalties optimal at different rates of divergence
Two major problems of Clustal :

1. **Local minimum problem** :
   - Greedy algorithm may get stuck in local minima, and mistakes get accumulated at each step.
   - Need iterative/stochastic procedures to solve this

2. **Choice of alignment parameters** :
   - Free parameters - weight matrix, gap opening and gap extension penalties.
   - For similar sequences, many possible values
   - Different weight matrices and gap penalties optimal at different rates of divergence

Addresses the second problem :
• Two major problems of Clustal:

1. **Local minimum problem**:  
   • Greedy algorithm may get stuck in local minima, and mistakes get accumulated at each step.  
   • Need iterative/stochastic procedures to solve this

2. **Choice of alignment parameters**:  
   • Free parameters - weight matrix, gap opening and gap extension penalties.  
   • For similar sequences, many possible values  
   • Different weight matrices and gap penalties optimal at different rates of divergence

• Addresses the second problem:
  
  • Choice of weight matrix: Dayhoff PAM / BLOSUM
Two major problems of Clustal:

1. **Local minimum problem:**
   - Greedy algorithm may get stuck in local minima, and mistakes get accumulated at each step.
   - Need iterative/stochastic procedures to solve this

2. **Choice of alignment parameters:**
   - Free parameters - weight matrix, gap opening and gap extension penalties.
   - For similar sequences, many possible values
   - Different weight matrices and gap penalties optimal at different rates of divergence

Addresses the second problem:

- Choice of weight matrix: Dayhoff PAM / BLOSUM
- Varying gap penalties:
Two major problems of Clustal:

1. **Local minimum problem**:
   - Greedy algorithm may get stuck in local minima, and mistakes get accumulated at each step.
   - Need iterative/stochastic procedures to solve this

2. **Choice of alignment parameters**:
   - Free parameters - weight matrix, gap opening and gap extension penalties.
   - For similar sequences, many possible values
   - Different weight matrices and gap penalties optimal at different rates of divergence

Addresses the second problem:

- Choice of weight matrix: Dayhoff PAM / BLOSUM
- Varying gap penalties:
  - Dependence on weight matrix, sequence lengths, difference in lengths of 2 sequences
Two major problems of Clustal:

1. **Local minimum problem**:
   - Greedy algorithm may get stuck in local minima, and mistakes get accumulated at each step.
   - Need iterative/stochastic procedures to solve this

2. **Choice of alignment parameters**:
   - Free parameters - weight matrix, gap opening and gap extension penalties.
   - For similar sequences, many possible values
   - Different weight matrices and gap penalties optimal at different rates of divergence

Addresses the second problem:

- Choice of weight matrix: Dayhoff PAM / BLOSUM
- Varying gap penalties:
  - Dependence on weight matrix, sequence lengths, difference in lengths of 2 sequences
  - Position-specific and residue-specific gap penalties
Clustal W - 1994

• Two major problems of Clustal:

1. **Local minimum problem**:  
   • Greedy algorithm may get stuck in local minima, and mistakes get accumulated at each step.  
   • Need iterative/stochastic procedures to solve this

2. **Choice of alignment parameters**:  
   • Free parameters - weight matrix, gap opening and gap extension penalties.  
   • For similar sequences, many possible values  
   • Different weight matrices and gap penalties optimal at different rates of divergence

• Addresses the second problem:

  • Choice of weight matrix: Dayhoff PAM / BLOSUM

  • Varying gap penalties:

    ✦ Dependence on weight matrix, sequence lengths, difference in lengths of 2 sequences  
    ✦ Position-specific and residue-specific gap penalties  
    ✦ Lowered penalties at existing gaps
Two major problems of Clustal:

1. **Local minimum problem**:
   - Greedy algorithm may get stuck in local minima, and mistakes get accumulated at each step.
   - Need iterative/stochastic procedures to solve this

2. **Choice of alignment parameters**:
   - Free parameters - weight matrix, gap opening and gap extension penalties.
   - For similar sequences, many possible values
   - Different weight matrices and gap penalties optimal at different rates of divergence

Addresses the second problem:

- **Choice of weight matrix**: Dayhoff PAM / BLOSUM

- **Varying gap penalties**:
  - Dependence on weight matrix, sequence lengths, difference in lengths of 2 sequences
  - Position-specific and residue-specific gap penalties
  - Lowered penalties at existing gaps
  - Increased penalties near existing gaps
Clustal W - 1994

• Two major problems of Clustal:

  1. **Local minimum problem**:
     • Greedy algorithm may get stuck in local minima, and mistakes get accumulated at each step.
     • Need iterative/stochastic procedures to solve this

  2. **Choice of alignment parameters**:
     • Free parameters - weight matrix, gap opening and gap extension penalties.
     • For similar sequences, many possible values
     • Different weight matrices and gap penalties optimal at different rates of divergence

• Addresses the second problem:

  • Choice of weight matrix: Dayhoff PAM / BLOSUM

  • Varying gap penalties:
    ♦ Dependence on weight matrix, sequence lengths, difference in lengths of 2 sequences
    ♦ Position-specific and residue-specific gap penalties
    ♦ Lowered penalties at existing gaps
    ♦ Increased penalties near existing gaps
    ♦ Reduced penalties in hydrophilic stretches
Clustal W

Some more changes:
- Pairwise distance matrix: NW/WL
- Guide tree: NJ
- Sequence weighting in progressive alignment
Clustal W

Some more changes:
• Pairwise distance matrix: NW/WL
• Guide tree: NJ
• Sequence weighting in progressive alignment
Clustal W

Some more changes:

- Pairwise distance matrix: NW/WL
- Guide tree: NJ
- Sequence weighting in progressive alignment

Pairwise distance matrix, unrooted NJ, rooted NJ with weighted sequences

The scoring scheme for comparing two positions from two alignments.
Multi clustal - 1999

- Automated method to find the optimal parameters in Clustal W
Multi clustal - 1999

- Automated method to find the optimal parameters in Clustal W

- Greedy approach consisting of 4 cycles:
Multi clustal - 1999

• Automated method to find the optimal parameters in Clustal W

• Greedy approach consisting of 4 cycles:
  1. Weight matrix varied and gap penalties fixed
Multi clustal - 1999

• Automated method to find the optimal parameters in Clustal W

• Greedy approach consisting of 4 cycles:

  1. Weight matrix varied and gap penalties fixed

  2. Weight matrix set to best value from step 1, gap opening penalties varied in small steps
Multi clustal - 1999

- Automated method to find the optimal parameters in Clustal W

- Greedy approach consisting of 4 cycles:
  1. Weight matrix varied and gap penalties fixed
  2. Weight matrix set to best value from step 1, gap opening penalties varied in small steps
  3. Wt. matrix, gap opening penalty fixed from 2 and 3, and gap extension penalty varied
Multi clustal - 1999

• Automated method to find the optimal parameters in Clustal W

• Greedy approach consisting of 4 cycles:

  1. Weight matrix varied and gap penalties fixed

  2. Weight matrix set to best value from step 1, gap opening penalties varied in small steps

  3. Wt. matrix, gap opening penalty fixed from 2 and 3, and gap extension penalty varied

  4. Gap penalties fixed from 2 and 3, and weight matrix again varied.
Multi clustal - 1999

- Automated method to find the optimal parameters in Clustal W

- Greedy approach consisting of 4 cycles:
  1. Weight matrix varied and gap penalties fixed
  2. Weight matrix set to best value from step 1, gap opening penalties varied in small steps
  3. Wt. matrix, gap opening penalty fixed from 2 and 3, and gap extension penalty varied
  4. Gap penalties fixed from 2 and 3, and weight matrix again varied.

- Best parameter choice in every cycle is the one which maximizes an alignment score:
Multi clustal - 1999

• Automated method to find the optimal parameters in Clustal W

• Greedy approach consisting of 4 cycles:
  
  1. Weight matrix varied and gap penalties fixed
  
  2. Weight matrix set to best value from step 1, gap opening penalties varied in small steps
  
  3. Wt. matrix, gap opening penalty fixed from 2 and 3, and gap extension penalty varied
  
  4. Gap penalties fixed from 2 and 3, and weight matrix again varied.

• Best parameter choice in every cycle is the one which maximizes an alignment score:

\[
\theta = (t+s)-(\gamma+(\alpha+\beta/2+\delta/4+\varepsilon/8))
\]

where \( \theta \) = alignment score, \( t \) = identical amino acids, \( \sigma \) = conservative amino acid substitutions, \( \gamma \) = gap events, and where: \( \alpha \) = # of single amino acid islands = -X-; where - is a gap and X is an amino acid; \( \beta \) = # of double amino acid islands = -XX-; \( \delta \) = # of triple amino acid islands = -XXX-; \( \varepsilon \) = # of quadruple amino acid island = -XXXX-.
Multi Clustal - Results

- Red boxes indicate misaligned residues or regions.
- Multiclustal introduces only three blocks of gaps in known loop structure regions.
- Clustal W has four blocks of gaps, and one block lies in between the beta sheet subdomain.
- Multi Clustal has only 2 misaligned residues, while Clustal W has 6
Clustal X - 1997
Clustal X - 1997

• Graphical User Interface for Clustal W
Clustal X - 1997

- Graphical User Interface for Clustal W
- Built on NCBI software development toolkit - portable across different OS
Clustal X - 1997

- Graphical User Interface for Clustal W
- Built on NCBI software development toolkit - portable across different OS
- Additional features: output alignment in color coded form, to allow user to detect the problem regions and realign them
Clustal X - 1997

- Graphical User Interface for Clustal W
- Built on NCBI software development toolkit - portable across different OS
- Additional features: output alignment in color coded form, to allow user to detect the problem regions and realign them
- Each column of each sequence and the alignment is embedded into an R dimensional space.
  - The embedding of the alignment is called the consensus sequence
  - Distance of the residue at position j in sequence i from the jth element of consensus sequence = $D_{ij}$
  - Exceptional residues: for which $D_{ij}$ is above some threshold
  - Low scoring segments: subsequences which have low matching score with other sequences
    ✓ Generally due to divergent sequences, frameshifts in input sequences, or misalignments
Clustal X - 1997

• Graphical User Interface for Clustal W

• Built on NCBI software development toolkit - portable across different OS

• Additional features: output alignment in color coded form, to allow user to detect the problem regions and realign them

• Each column of each sequence and the alignment is embedded into an R dimensional space.
  • The embedding of the alignment is called the consensus sequence
  • Distance of the residue at position j in sequence i from the jth element of consensus sequence = D_ij
  • Exceptional residues: for which D_ij is above some threshold
  • Low scoring segments: subsequences which have low matching score with other sequences
    ✓ Generally due to divergent sequences, frameshifts in input sequences, or misalignments

• Exceptional residues and low scoring segments displayed in different colors. User can select sequences or subsequences to realign
Clustal X - Results
DbClustal - 2007
• **Hybrid method** which combines local conservation information and global alignment to provide accurate alignment of highly divergent sequence set
**Hybrid method** which combines local conservation information and global alignment to provide accurate alignment of highly divergent sequence set

- Uses Ballast to create anchor points for local conservation

  - Ballast finds subsequences of query sequence which match subsequences of the database sequences, and gives it a score/weight.

    **Anchor File**:  
    
    \[\text{Seq: Name1 Name2 pos: P beg: R1 R2 len : L weight : W}\]

- These anchors represent locally conserved subsequences. We expect locally conserved regions more likely to be aligned in the final alignment than non-conserved regions.
DbClustal

• Incorporating anchors in global alignment: modify the residue alignment score $S_{i,j}$

$$ANCHOR(S1, S2)_{i,j} = \max_{k=1, \ldots, L}(0, W_k)$$

where $W_k$ is the weight for the $k^{th}$ anchor with the residue pairs $S1_i$ and $S2_j$. Thus,

$$S_{i,j} = C_{i,j} + ANCHOR_{i,j}$$

Similarly, for a group of sequences A and B

$$S_{i,j} = P_{i,j} + ANCHOR_{i,j}$$

where $P_{i,j}$ is the profile-to-profile alignment score for $A_i$ and $B_j$. 
Figure 3. A plot of the ratio of the MD scores for 1683 alignments obtained by ClustalW and DbClustal. The red lines denote the score ratio threshold of 1.05, above which the two alignments are considered to be significantly different.

MD : Mean Distance of the sequences from the alignment
Clustal Omega - 2011
• Clustal Omega is capable of performing fast alignments on virtually any number of sequences of any length
Clustal Omega - 2011

- Clustal Omega is capable of performing fast alignments on virtually any number of sequences of any length

- Uses mBed for clustering, which is $O(N \log N)$, unlike UPGMA, NJ which are $O(N^2)$
Clustal Omega - 2011

- Clustal Omega is capable of performing fast alignments on virtually any number of sequences of any length

- Uses **mBed** for clustering, which is $O(N\log N)$, unlike UPGMA, NJ which are $O(N^2)$

- **EPA** - It uses External Profile HMMs of pre-aligned sequences from the same family as the input sequences.
Clustal Omega - 2011

- Clustal Omega is capable of performing fast alignments on virtually any number of sequences of any length

- Uses **mBed** for clustering, which is $O(N\log N)$, unlike UPGMA, NJ which are $O(N^2)$

- **EPA** - It uses External Profile HMMs of pre-aligned sequences from the same family as the input sequences.

- The input sequences are aligned to the EP instead of being aligned to each other
Clustal Omega - Results

On small datasets:

Table 1 BAliBASE results

<table>
<thead>
<tr>
<th>Aligner</th>
<th>Av score (218 families)</th>
<th>BB1 (38 families)</th>
<th>BB12 (44 families)</th>
<th>BB2 (41 families)</th>
<th>BB3 (30 families)</th>
<th>BB4 (49 families)</th>
<th>BB5 (16 families)</th>
<th>Tot time (s)</th>
<th>Consistency</th>
</tr>
</thead>
<tbody>
<tr>
<td>MSAProbs</td>
<td>0.607</td>
<td>0.441</td>
<td>0.865</td>
<td>0.464</td>
<td>0.607</td>
<td>0.622</td>
<td>0.608</td>
<td>12382.00</td>
<td>Yes</td>
</tr>
<tr>
<td>Probalign</td>
<td>0.589</td>
<td>0.453</td>
<td>0.862</td>
<td>0.439</td>
<td>0.566</td>
<td>0.603</td>
<td>0.549</td>
<td>10095.20</td>
<td>Yes</td>
</tr>
<tr>
<td>MAFFT (auto)</td>
<td>0.588</td>
<td>0.439</td>
<td>0.831</td>
<td>0.450</td>
<td>0.581</td>
<td>0.605</td>
<td>0.591</td>
<td>1475.40</td>
<td>Mostly (203/218)</td>
</tr>
<tr>
<td>Procons</td>
<td>0.558</td>
<td>0.417</td>
<td>0.855</td>
<td>0.406</td>
<td>0.544</td>
<td>0.532</td>
<td>0.573</td>
<td>13086.30</td>
<td>Yes</td>
</tr>
<tr>
<td>Clustal Ω</td>
<td>0.554</td>
<td>0.358</td>
<td>0.789</td>
<td>0.450</td>
<td>0.575</td>
<td>0.579</td>
<td>0.533</td>
<td>539.91</td>
<td>No</td>
</tr>
<tr>
<td>T-Coffee</td>
<td>0.551</td>
<td>0.410</td>
<td>0.848</td>
<td>0.402</td>
<td>0.491</td>
<td>0.545</td>
<td>0.587</td>
<td>81041.50</td>
<td>Yes</td>
</tr>
<tr>
<td>Kalign</td>
<td>0.501</td>
<td>0.365</td>
<td>0.790</td>
<td>0.360</td>
<td>0.476</td>
<td>0.504</td>
<td>0.435</td>
<td>21.88</td>
<td>No</td>
</tr>
<tr>
<td>MUSCLE</td>
<td>0.475</td>
<td>0.318</td>
<td>0.804</td>
<td>0.350</td>
<td>0.409</td>
<td>0.450</td>
<td>0.460</td>
<td>789.57</td>
<td>No</td>
</tr>
<tr>
<td>MAFFT (default)</td>
<td>0.458</td>
<td>0.258</td>
<td>0.749</td>
<td>0.316</td>
<td>0.425</td>
<td>0.480</td>
<td>0.496</td>
<td>68.24</td>
<td>No</td>
</tr>
<tr>
<td>FSA</td>
<td>0.419</td>
<td>0.270</td>
<td>0.818</td>
<td>0.187</td>
<td>0.259</td>
<td>0.474</td>
<td>0.398</td>
<td>53648.10</td>
<td>No</td>
</tr>
<tr>
<td>Dialign</td>
<td>0.415</td>
<td>0.265</td>
<td>0.696</td>
<td>0.292</td>
<td>0.312</td>
<td>0.441</td>
<td>0.425</td>
<td>3977.44</td>
<td>No</td>
</tr>
<tr>
<td>PRANK</td>
<td>0.376</td>
<td>0.223</td>
<td>0.680</td>
<td>0.257</td>
<td>0.321</td>
<td>0.360</td>
<td>0.356</td>
<td>128355.00</td>
<td>No</td>
</tr>
<tr>
<td>ClustalW</td>
<td>0.374</td>
<td>0.227</td>
<td>0.712</td>
<td>0.220</td>
<td>0.272</td>
<td>0.396</td>
<td>0.308</td>
<td>766.47</td>
<td>No</td>
</tr>
</tbody>
</table>
On large datasets:

<table>
<thead>
<tr>
<th>Aligner</th>
<th>0 ≤ %ID ≤ 100 (1682 families)</th>
<th>0 ≤ %ID ≤ 20 (912 families)</th>
<th>20 ≤ %ID ≤ 40 (563 families)</th>
<th>40 ≤ %ID ≤ 70 (117 families)</th>
<th>70 ≤ %ID ≤ 100 (90 families)</th>
<th>Total time (s) (1682 families)</th>
</tr>
</thead>
<tbody>
<tr>
<td>MSAprobs</td>
<td>0.737</td>
<td>0.591</td>
<td>0.889</td>
<td>0.965</td>
<td>0.971</td>
<td>51 286.00</td>
</tr>
<tr>
<td>MAFFT</td>
<td>0.721</td>
<td>0.569</td>
<td>0.876</td>
<td>0.961</td>
<td>0.979</td>
<td>45 444.45</td>
</tr>
<tr>
<td>(auto)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ProBalign</td>
<td>0.719</td>
<td>0.563</td>
<td>0.881</td>
<td>0.961</td>
<td>0.977</td>
<td>35 117.30</td>
</tr>
<tr>
<td>Probcons</td>
<td>0.717</td>
<td>0.562</td>
<td>0.876</td>
<td>0.955</td>
<td>0.972</td>
<td>46 908.30</td>
</tr>
<tr>
<td>T-Coffee</td>
<td>0.710</td>
<td>0.558</td>
<td>0.865</td>
<td>0.950</td>
<td>0.972</td>
<td>175 789.00</td>
</tr>
<tr>
<td>Clustal Ω</td>
<td>0.700</td>
<td>0.535</td>
<td>0.866</td>
<td>0.967</td>
<td>0.980</td>
<td>1698.06</td>
</tr>
<tr>
<td>MUSCLE</td>
<td>0.677</td>
<td>0.507</td>
<td>0.850</td>
<td>0.946</td>
<td>0.976</td>
<td>2068.56</td>
</tr>
<tr>
<td>MAFFT</td>
<td>0.677</td>
<td>0.513</td>
<td>0.836</td>
<td>0.961</td>
<td>0.979</td>
<td>225.56</td>
</tr>
<tr>
<td>Kalign</td>
<td>0.649</td>
<td>0.474</td>
<td>0.817</td>
<td>0.957</td>
<td>0.979</td>
<td>80.81</td>
</tr>
<tr>
<td>ClustalW2</td>
<td>0.617</td>
<td>0.430</td>
<td>0.797</td>
<td>0.933</td>
<td>0.975</td>
<td>3433.53</td>
</tr>
<tr>
<td>Dialign</td>
<td>0.595</td>
<td>0.398</td>
<td>0.783</td>
<td>0.940</td>
<td>0.974</td>
<td>18 909.70</td>
</tr>
<tr>
<td>PRANK</td>
<td>0.586</td>
<td>0.390</td>
<td>0.767</td>
<td>0.951</td>
<td>0.978</td>
<td>351 498.00</td>
</tr>
<tr>
<td>FSA</td>
<td>0.534</td>
<td>0.277</td>
<td>0.791</td>
<td>0.965</td>
<td>0.976</td>
<td>229 391.00</td>
</tr>
</tbody>
</table>
Clustal Omega - Results

Effect of using EPA

A: ClustalΩ HomFam

B: ClustalΩ BAliBASE
<table>
<thead>
<tr>
<th>Method</th>
<th>Language</th>
<th>Major changes / new features</th>
<th>Pairwise distance</th>
<th>Guide tree</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clustal</td>
<td>Fortran</td>
<td>-</td>
<td>WL</td>
<td>UPGMA</td>
</tr>
<tr>
<td>Clustal V</td>
<td>C</td>
<td>Flexible sequence input and alignment output format, ability to store and reuse old alignments, calculate phylogenetic trees with bootstrap confidence intervals after alignment</td>
<td>WL</td>
<td>UPGMA</td>
</tr>
<tr>
<td>Clustal W</td>
<td>C</td>
<td>Fast and accurate alignment for highly divergent sequences. Choice of alignment parameters: varying gap penalties and weight matrices; sequence weighting to obtain progressive alignment score.</td>
<td>WL / NW</td>
<td>NJ</td>
</tr>
<tr>
<td>Multi Clustal</td>
<td>Perl</td>
<td>Automated multi-cycle greedy method for finding better alignment parameters in Clustal W. Use of a quantitative alignment score for choosing the best alignment parameters.</td>
<td>Same as Clustal W</td>
<td>Same as Clustal W</td>
</tr>
<tr>
<td>Clustal X</td>
<td>C</td>
<td>Graphical interface for Clustal W; defining a scheme for identifying exceptional residues and low-scoring segments color-coding such regions, so user can select some sequences or regions to re-align.</td>
<td>Same as clustal W</td>
<td>Same as Clustal W</td>
</tr>
<tr>
<td>Clustal W, X v 2.0</td>
<td>C++</td>
<td>Iterative alignment at the final stage or during the progressive alignment to refine the alignments further.</td>
<td>Same as clustal W</td>
<td>UPGMA/NJ</td>
</tr>
<tr>
<td>DbClustal</td>
<td>C</td>
<td>Incorporation of local conservation information with global alignment. Uses Ballast anchors to modify the residue alignment score to favour alignment of locally conserved regions.</td>
<td>Same as Clustal W, with modified residue alignment score for local conservation (anchor score added to weight matrix score)</td>
<td>Same as Clustal W</td>
</tr>
<tr>
<td>Clustal Omega</td>
<td>C, C++</td>
<td>Used mBed for fast clustering, and External Profiles for more accurate alignments.</td>
<td>No pairwise distance calculation.</td>
<td>mBed clustering</td>
</tr>
</tbody>
</table>
Takeaways
Takeaways

• Clustal family of alignments has grown a lot since 1988
Takeaways

• Clustal family of alignments has grown a lot since 1988

• Most work was done in the direction of making the system user-interactive, allowing flexible input and output formats, increasing portability across different systems, increasing alignment accuracy on highly divergent sequences, and making the algorithm scale well to large datasets
Takeaways

• Clustal family of alignments has grown a lot since 1988

• Most work was done in the direction of making the system user-interactive, allowing flexible input and output formats, increasing portability across different systems, increasing alignment accuracy on highly divergent sequences, and making the algorithm scale well to large datasets.

• To achieve these aims, various levels of the algorithm were experimented with - different pairwise distance methods, guide tree construction methods, and progressive alignment methods.
• Clustal family of alignments has grown a lot since 1988

• Most work was done in the direction of making the system user-interactive, allowing flexible input and output formats, increasing portability across different systems, increasing alignment accuracy on highly divergent sequences, and making the algorithm scale well to large datasets.

• To achieve these aims, various levels of the algorithm were experimented with - different pairwise distance methods, guide tree construction methods, and progressive alignment methods.

• More control was given to user to identify problematic regions or manually refine the alignments.
Takeaways

• Clustal family of alignments has grown a lot since 1988

• Most work was done in the direction of making the system user-interactive, allowing flexible input and output formats, increasing portability across different systems, increasing alignment accuracy on highly divergent sequences, and making the algorithm scale well to large datasets.

• To achieve these aims, various levels of the algorithm were experimented with - different pairwise distance methods, guide tree construction methods, and progressive alignment methods.

• More control was given to user to identify problematic regions or manually refine the alignments.

• Methods for exploring the problem of finding the optimal free parameters heuristically or automatically; Formulation of alignment quality scores to judge the quality in absence of gold alignments.
Takeaways

- Clustal family of alignments has grown a lot since 1988

- Most work was done in the direction of making the system user-interactive, allowing flexible input and output formats, increasing portability across different systems, increasing alignment accuracy on highly divergent sequences, and making the algorithm scale well to large datasets.

- To achieve these aims, various levels of the algorithm were experimented with - different pairwise distance methods, guide tree construction methods, and progressive alignment methods.

- More control was given to user to identify problematic regions or manually refine the alignments.

- Methods for exploring the problem of finding the optimal free parameters heuristically or automatically; Formulation of alignment quality scores to judge the quality in absence of gold alignments.

- Local conservation information was incorporated with global alignment to get more accurate alignments. External Profile HMMs and multiple iterations also helped increase accuracies.
Takeaways

- Clustal family of alignments has grown a lot since 1988

- Most work was done in the direction of making the system user-interactive, allowing flexible input and output formats, increasing portability across different systems, increasing alignment accuracy on highly divergent sequences, and making the algorithm scale well to large datasets.

- To achieve these aims, various levels of the algorithm were experimented with - different pairwise distance methods, guide tree construction methods, and progressive alignment methods.

- More control was given to user to identify problematic regions or manually refine the alignments.

- Methods for exploring the problem of finding the optimal free parameters heuristically or automatically; Formulation of alignment quality scores to judge the quality in absence of gold alignments.

- Local conservation information was incorporated with global alignment to get more accurate alignments. External Profile HMMs and multiple iterations also helped increase accuracies.

- To scale to very large datasets, pairwise distance computation was avoided and fast clustering was used to get the alignment order.
Thanks! :)