Bioinformatics I
WS 15/16
Tutor: Alexander Seitz

1	2	3	$\sum$

Jonas Ditz
& Benjamin Schroeder

Assignment 4

(Abgabe am 9. November 2015)

Theoretical Assignment - *Optimal multiple alignment*

To calculate our MSA we use the recursion written down on page 50 in the script.

$$F(i_1,i_2,i_3) = max \begin{cases} F(i_1-1,i_2-1,i_3-1) + s_{SP}(a_{1i_1},a_{2i_2},a_{3i_3}) \\ \\ F(i_1-1,i_2-1,i_3) + s_{SP}(a_{1i_1},a_{2i_2},-) \\ F(i_1-1,i_2,i_3-1) + s_{SP}(a_{1i_1},-,a_{3i_3}) \\ F(i_1,i_2-1,i_3-1) + s_{SP}(-,a_{2i_2},a_{3i_3}) \\ \\ F(i_1-1,i_2,i_3) + s_{SP}(a_{1i_1},-,-) \\ F(i_1,i_2-1,i_3) + s_{SP}(-,a_{2i_2},-) \\ F(i_1,i_2,i_3-1) + s_{SP}(-,-,a_{3i_3}) \end{cases}$$

Such a recursion results in a three-dimensional matrix. Since such a matrix is really difficult to sketch, we split it into three two-dimensional matrices.

0	0	C	C	T
0	**0**	-2	-4	-6
C	-2	**-2**	-4	-10
T	-4	-8	-8	-6
T	-6	-10	-12	-10

T	0	C	C	T
0	-2	-4	-6	-8
C	-4	-4	-6	-10
T	-6	-8	**-4**	2
T	-8	-10	-8	2

C	0	C	C	T
0	-4	-6	-8	-10
C	-6	2	2	-8
T	-8	-6	-2	0
T	-10	-8	-6	**-6**

Figure 1: DP matrix of MSA, traceback is shown in red

If we fill the DP matrix using this recursion, we get several optimal alignments. One of them is the following:

$$\begin{pmatrix} C & T & T \\ - & T & C \\ C & C & T \end{pmatrix} \tag{1}$$

with score $\alpha_{SP}(A^*) = S(A,B) + S(A,C) + S(B,C) = -2 + 2 + (-6) = -6$.

Theoretical Assignment - *Progressive alignment*

Progressive alignment methodes are one way to create multiple sequence alignments (MSA), in this task we had to accomplish manualy a progressive alignment. The first step for the progressive alignment was to generate the pairwise global distance matrices. Afterwards the global optima were entered into a new table, which was used to generate the guide tree in the second step.(figure 2).

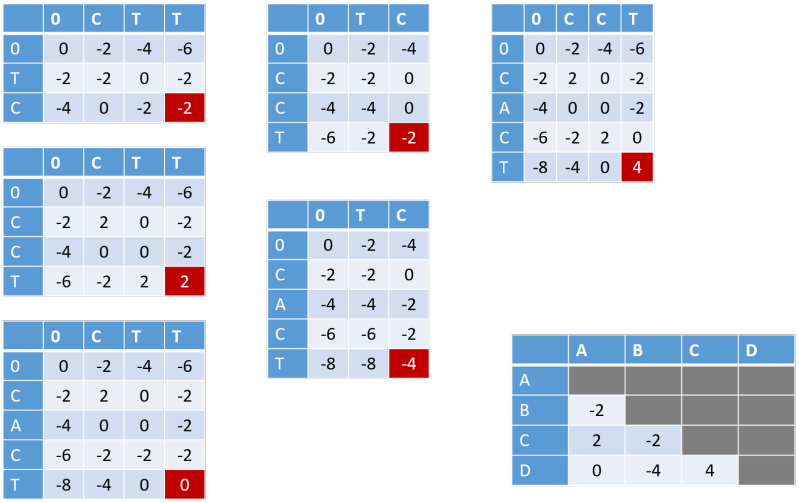

Figure 2: Global alignments lead to the basis table used to generate the guide tree

In step 2 the guide tree was created by the UPGMA methode in 3 substeps. After every step, which added a new cluster, the distance table was updated. The result from this step was the guide tree, which was used to apply the 'complete alignment' methode in the last step, the progressive alignment step.(figure 3)

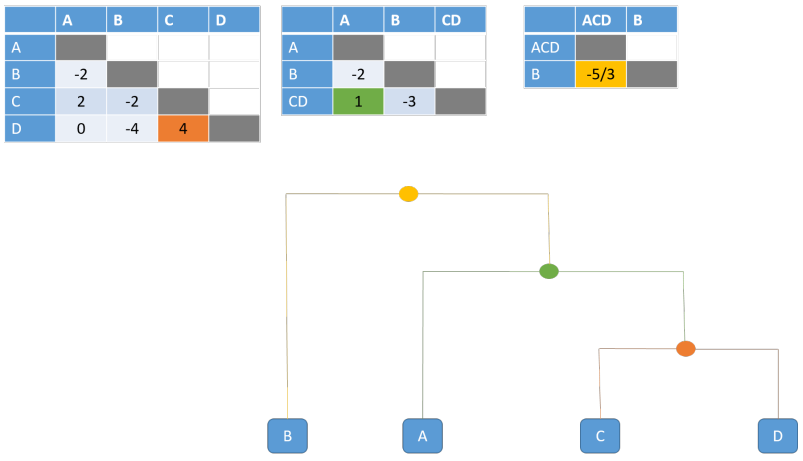

Figure 3: The guide tree is made from the final table from step 1

The last step was creating the actual MSA. The 'complete alignment' methode was used. This means after every step, the distance of the new sequences was determined to each other sequence in the cluster. The optimal resulting score was used to get the best alignment for the new Sequence.

after 3 steps the MSA was completed. (figure 4)

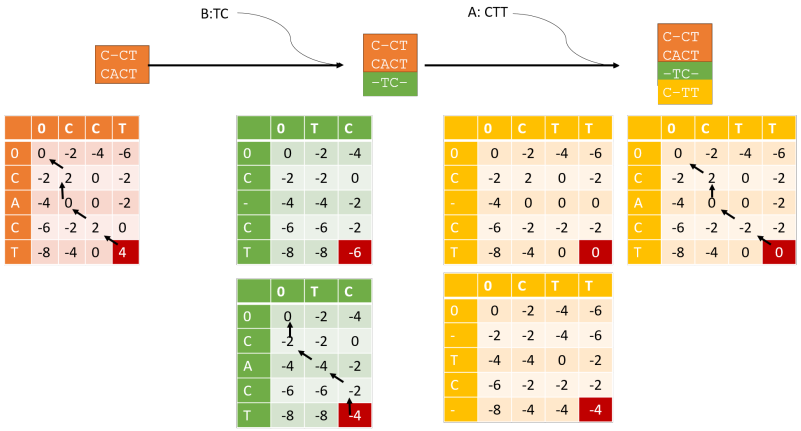

Figure 4: Complete alignment methode is used to get an MSA from the guidetree from step 2

Practical Assignment - *Comparing multiple alignment*

Multiple sequence alignments (MSA) are very important to detect similarities between several sequences and e.g. determine whether a sequence belongs to a certain family or not. Hence, many different programs, such as ClustalW, were developed to calculate a MSA. We calculated a multiple sequence alignment for three files (BB11007, BB20002 and Prolyl-tRNA), each file contained several sequences, with different programs to compare the results. The used programs are ClustalΩ [**?**], MAFFT [**?**] and MUSCLE [**?**, **?**]. The resulting MSAs were compared with a gold standard provided by BaliBase [**?**]. This database provide a large number of reference MSAs as well as a program "BaliScore", which can be used to calculate different scores to evaluate your MSA. For this assignment we are interested in the total column score, which is the percentage of accurately reconstructed columns of the reference MSA, and the sum-of-pair score, which is for BaliScore the percentage of pairs of aligned residues that are similar in the reference and the reconstructed MSA. Furthermore, a important information is the runtime of different programs. We used the bash command *time* to collect this information.

ClustalΩ [**?**] is the newest extension for the Clustal family. One problem of ClustalW was the scalability. With ClustalΩ a MSA can now calculated for hundreds of thousands of sequences within a few hours. On the technical side the algorithm can use multiple processors. According to the official site[1], ClustalΩ produces better alignments than previous versions. MAFFT [**?**] offers several modes. The L-INS-i mode is slower and only suitable for alignments with less than 200 sequences but results in a more accurate alignment. Contrary to that mode the FFT-NS-2 mode can be used to align up to 300.000 sequences. It is a lot faster by using the Fast Fourier Transformation but also less accurate. We used the FFT-NS-2 mode for this assignment. The third program MUSCLE [**?**] his a iterative approach to multiple sequence alignments. It uses a very accurate distance measure two calculate relatedness of two sequences and updates this distance

[1] http://www.clustal.org/omega/

measure for every iteration step.

The resulting total column scores can be found in figure 5. All calculated sum- of-pairs scores are written down in figure 6 and the runtime of all programs can be found in figure 7. One can see that the total column score as well as the sum-of-pairs score for Prolyl-tRNA is missing. Although we calculated the MSAs with all three programs for these sequences, BaliScore was not able to calculate a score. We got a memory error, even with five Gb accessible memory. Also interesting is the total column score for the second biggest file, BB20002. BaliScore calculated a score of 0.00 for all of the three programms. If we assume that the program calculated valid scores, one can see that the gold standard provided by BaliBase differs a lot compared to the calculated MSAs. Even for the smallest file, BB11007, the total column score barely reaches 0.4. That means only 40 percent of the columns in the reference MSA were identically reproduced by our used algorithms. Since all three programs are heuristics, it is not surprising that the calculated MSAs are not similar with the gold standard. One can explain such a big difference with the occurrence of several optimal MSAs of the sequences. And the algorithms have calculated another MSA than the gold standard. Another explanation for our results would be that the heuristics worked worst than expected. By comparing runtimes of ClustalΩ, MUSCLE and MAFFT one can see that MUSCLE is by far the slowest program. This is logical, since MUSCLE is a iterative approach to multiple sequence alignment. But our results for MUSCLE are not significant better than for the other algorithms. So in that case we paid with a lot of computational time for no improvement.

total column score	**ClustalΩ**	**MUSCLE**	**MAFFT**
BB11007	0.33	0.39	0.24
BB20002	0.00	0.00	0.00
Prolyl-tRNA	-	-	-

Figure 5: total column score of all alignments

SP-score	**ClustalΩ**	**MUSCLE**	**MAFFT**
BB11007	0.597	0.582	0.524
BB20002	0.314	0.197	0.246
Prolyl-tRNA	-	-	-

Figure 6: sum-of-pairs score of all alignments

runtime	**ClustalΩ**	**MUSCLE**	**MAFFT**
BB11007	0.389	0.488	0.13
BB20002	6.62	19.928	0.38
Prolyl-tRNA	16.11	4116.97	7.72

Figure 7: runtime of all used programs for example files (in sec)

References