**Course: Advanced Bioinformatics**

**Module title: Global Alignment**

**Module no. : 30**

**Dynamic Programming**

Luckily, sequence alignment has an optimal-substructure property, and therefore there is a much easier way to consider all of the possible alignments using what is called dynamic programming (DP). Dynamic programming techniques are used in many different aspects of computer science. DP algorithms solve optimization problems by dividing the problem into independent sub problems. Each sub problem is then only stored once, and the answer is stored in a table, thus avoiding the work of re-computing the solution.

With sequence alignment, the sub problems can be thought of as the alignment of the “prefixes” of the two sequences to a certain point. Therefore, a dynamic programming matrix is computed. The optimal alignment score for any particular point in the matrix is built upon the optimal alignment that has been computed to that point.

Dynamic programming techniques align two sequences by beginning at the ends of the two sequences and attempting to align all possible pairs of characters (one from each sequence) using a scoring scheme for matches, mismatches, and gaps. The highest set of scores defines the optimal alignment between the two sequences.

We will first consider dynamic programming in terms of DNA, where only exact matches are considered for a match score. Later we will discuss how substitution matrices can be used to score amino acid matches and mismatches.

Dynamic programming approaches are guaranteed to provide the optimal alignment given a particular scoring scheme. For large sequences, dynamic programming can be slow and memory intensive. Discuss the time and space necessary for microarray analysis.

**Setting up the Dynamic Programming Matrix**

Now we are ready to go ahead and start creating the dynamic programming matrix. The first step is to align one of the sequences across the columns of the matrix, and the other sequence across the rows. Note that an alignment can also begin with a gap in one of the sequences, so that has to be taken care of as well. Let’s assume that we want to align the sequence GAATTCAGTTA to GGATCGA. The length of the first sequence is 11 residues, and the length of the second is 7. Since it is possible to begin an alignment with a gap, the size of the matrix should be 8 x 12. Row 0 and column 0 will represent gaps. Rows 1-7 will be labeled with the corresponding residue of the sequence GGATCGA, while columns 1-11 will be labeled with the corresponding residue of the sequence GAATTCAGTTA. The initial matrix, ***S***, is as follows:

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **-** | **G** | **A** | **A** | **T** | **T** | **C** | **A** | **G** | **T** | **T** | **A** |
| **-** |  |  |  |  |  |  |  |  |  |  |  |  |
| **G** |  |  |  |  |  |  |  |  |  |  |  |  |
| **G** |  |  |  |  |  |  |  |  |  |  |  |  |
| **A** |  |  |  |  |  |  |  |  |  |  |  |  |
| **T** |  |  |  |  |  |  |  |  |  |  |  |  |
| **C** |  |  |  |  |  |  |  |  |  |  |  |  |
| **G** |  |  |  |  |  |  |  |  |  |  |  |  |
| **A** |  |  |  |  |  |  |  |  |  |  |  |  |

Now we need to decide upon the scoring scheme to be used. This requires parameters for a match score, a mismatch score, and a gap score. The match and mismatch scores will be combined into a single match/mismatch score, s(aibj). We’ll see how this can later be used with a substitution matrix. There will also be a single linear gap penalty score, w. For our first example, we have the following parameters:

Sequence #1: GAATTCAGTTA; M = 11

Sequence #2: GGATCGA; N = 7

* s(aibj) = +5 if ai = bj (match score)
* s(aibj) = -3 if ai≠bj (mismatch score)
* w = -4 (gap penalty)

**Three steps in dynamic programming**

Once you have the scoring functions set and the sequences to align, there are three steps involved in calculating the optimal scoring alignment. The methods used to finish these three steps are dependent upon whether global or local sequence alignment is desired. The three steps are as follows:

* Initialization
* Matrix Fill (scoring)
* Traceback (alignment)

***Global Alignment: Needleman-Wunsch Algorithm***

In global sequence alignment, an attempt to align the entirety of two different sequences is made, up to and including the ends of the sequence. Needleman and Wunsch (1970) were among the first to describe a dynamic programming algorithm for global sequence alignment.

**Initialization Step.** In the initialization step of global alignment, each row Si,0 is set to w \* i. In addition, each column S0,j is set to w \* j. Remember, that w is the gap penalty. Using the scoring scheme described above, the initialization step results in the following:

**Matrix Fill Step**. One possible solution of the matrix fill step finds the maximum global alignment score by starting in the upper left hand corner in the matrix and finding the maximal score Si,j for each position in the matrix. In order to find Si,j for any i,j it is minimal to know the score for the matrix positions to the left, above and diagonal to i, j. In terms of matrix positions, it is necessary to know Si-1,j, Si,j-1 and Si-1, j-1.

For each position, Si,j is defined to be the maximum score at position i,j; i.e.

**Si,j = MAXIMUM[**

**Si-1, j-1 + s(ai,bj**) (match/mismatch in the diagonal),

**Si,j-1 + w** (gap in sequence #1),

**Si-1,j + w** (gap in sequence #2)**]**

Note that in the example, Si-1,j-1 will be red, Si,j-1 will be green and Si-1,j will be blue.

Using this information, the score at position 1,1 in the matrix can be calculated. Since the first residue in both sequences is a G, s(a1b1)= 5, and by the assumptions stated earlier, w = -4. Thus, S1,1 = MAX[S0,0 + 5, S1,0 - 4, S0,1 - 4] = MAX[5, -8, -8].

A value of 5 is then placed in position 1,1 of the scoring matrix. Note that there is also an arrow placed back into the cell that resulted in the maximum score, S0,0.

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **-** | **G** | **A** | **A** | **T** | **T** | **C** | **A** | **G** | **T** | **T** | **A** |
| **-** | **0** | **-4** | **-8** | **-12** | **-16** | **-20** | **-24** | **-28** | **-32** | **-36** | **-40** | **-44** |
| **G** | **-4** | **5** |  |  |  |  |  |  |  |  |  |  |
| **G** | **-8** |  |  |  |  |  |  |  |  |  |  |  |
| **A** | **-12** |  |  |  |  |  |  |  |  |  |  |  |
| **T** | **-16** |  |  |  |  |  |  |  |  |  |  |  |
| **C** | **-20** |  |  |  |  |  |  |  |  |  |  |  |
| **G** | **-24** |  |  |  |  |  |  |  |  |  |  |  |
| **A** | **-28** |  |  |  |  |  |  |  |  |  |  |  |

Now we proceed to S1,2. Since a1 = G and b2 = A, there is a mismatch. Therefore, sa1b2 = -3 and by the assumptions stated earlier, w = -4. Thus, S1,2 = MAX[S0,1 -3, S1,1 - 4, S0,2 - 4] = MAX[-4 - 3, 5 – 4, -8 – 4] = MAX[-7, 1, -12] = 1. An arrow is placed back into the cell that resulted in the maximum score, which is the cell S1,1.

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **-** | **G** | **A** | **A** | **T** | **T** | **C** | **A** | **G** | **T** | **T** | **A** |
| **-** | **0** | **-4** | **-8** | **-12** | **-16** | **-20** | **-24** | **-28** | **-32** | **-36** | **-40** | **-44** |
| **G** | **-4** | **5** | 1 |  |  |  |  |  |  |  |  |  |
| **G** | **-8** |  |  |  |  |  |  |  |  |  |  |  |
| **A** | **-12** |  |  |  |  |  |  |  |  |  |  |  |
| **T** | **-16** |  |  |  |  |  |  |  |  |  |  |  |
| **C** | **-20** |  |  |  |  |  |  |  |  |  |  |  |
| **G** | **-24** |  |  |  |  |  |  |  |  |  |  |  |
| **A** | **-28** |  |  |  |  |  |  |  |  |  |  |  |

We can proceed to fill in the rest of the first row in a similar fashion, resulting in the following matrix:

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **-** | **G** | **A** | **A** | **T** | **T** | **C** | **A** | **G** | **T** | **T** | **A** |
| **-** | **0** | **-4** | **-8** | **-12** | **-16** | **-20** | **-24** | **-28** | **-32** | **-36** | **-40** | **-44** |
| **G** | **-4** | **5** | 1 | -3 | -7 | -11 | -15 | -19 | -23 | -27 | -31 | -35 |
| **G** | **-8** |  |  |  |  |  |  |  |  |  |  |  |
| **A** | **-12** |  |  |  |  |  |  |  |  |  |  |  |
| **T** | **-16** |  |  |  |  |  |  |  |  |  |  |  |
| **C** | **-20** |  |  |  |  |  |  |  |  |  |  |  |
| **G** | **-24** |  |  |  |  |  |  |  |  |  |  |  |
| **A** | **-28** |  |  |  |  |  |  |  |  |  |  |  |

Now we can start to fill in the second row, beginning with S2,1. Note that a2 = G and b1 = G, so sa2b1 = 5 and by the assumptions stated earlier, w = -4. Thus, S2,1= MAX[S1,0 +5, S0,2 - 4, S1,1 - 4] = MAX-4 + 5, -8 – 4, 5 - 4] = MAX[1, -12, 1] = 1. Note that in this case, there are two possible paths to the maximum value. Therefore, an arrow is placed back into each cell resulting in the maximum score, which are sells S1,0 and S1,1.

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **-** | **G** | **A** | **A** | **T** | **T** | **C** | **A** | **G** | **T** | **T** | **A** |
| **-** | **0** | **-4** | **-8** | **-12** | **-16** | **-20** | **-24** | **-28** | **-32** | **-36** | **-40** | **-44** |
| **G** | **-4** | **5** | 1 | -3 | -7 | -11 | -15 | -19 | -23 | -27 | -31 | -35 |
| **G** | **-8** | **1** |  |  |  |  |  |  |  |  |  |  |
| **A** | **-12** |  |  |  |  |  |  |  |  |  |  |  |
| **T** | **-16** |  |  |  |  |  |  |  |  |  |  |  |
| **C** | **-20** |  |  |  |  |  |  |  |  |  |  |  |
| **G** | **-24** |  |  |  |  |  |  |  |  |  |  |  |
| **A** | **-28** |  |  |  |  |  |  |  |  |  |  |  |

We can then proceed to fill in the rest of the matrix in a similar fashion. The resulting matrix is as follows:

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **-** | **G** | **A** | **A** | **T** | **T** | **C** | **A** | **G** | **T** | **T** | **A** |
| **-** | **0** | **-4** | **-8** | **-12** | **-16** | **-20** | **-24** | **-28** | **-32** | **-36** | **-40** | **-44** |
| **G** | **-4** | **5** | 1 | -3 | -7 | -11 | -15 | -19 | -23 | -27 | -31 | -35 |
| **G** | **-8** | **1** | 2 | -2 | -6 | -10 | -14 | -18 | -14 | -18 | -22 | -26 |
| **A** | **-12** | -3 | 6 | 7 | 3 | -1 | -5 | -9 | -13 | -17 | -21 | -17 |
| **T** | **-16** | -7 | 2 | 3 | 12 | 8 | 4 | 0 | -4 | -8 | -12 | -16 |
| **C** | **-20** | -11 | -2 | -1 | 8 | 9 | 13 | 9 | 5 | 1 | -3 | -7 |
| **G** | **-24** | -15 | -6 | -5 | 4 | 5 | 9 | 10 | 14 | 10 | 6 | 2 |
| **A** | **-28** | -19 | -10 | -1 | 0 | 1 | 5 | 14 | 10 | 11 | 7 | 11 |

Each cell has one to three arrows indicating from which cell the maximum score was obtained. The matrix fill step is now complete.

Traceback Step. After the matrix fill step, the maximum global alignment score for the two sequences is 11 (the value in the lower right hand cell). The traceback step will obtain the actual alignment(s) that result in the maximum score.

The traceback begins in position SM,N; i.e. the position where both sequences are globally aligned.

Since pointers have been kept back to all possible predacessors, the traceback is simple. At each cell, we look to see where we move next according to the pointers. To begin, the only possible predacessor is the diagonal match.

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **-** | **G** | **A** | **A** | **T** | **T** | **C** | **A** | **G** | **T** | **T** | **A** |
| **-** | **0** | **-4** | **-8** | **-12** | **-16** | **-20** | **-24** | **-28** | **-32** | **-36** | **-40** | **-44** |
| **G** | **-4** | **5** | 1 | -3 | -7 | -11 | -15 | -19 | -23 | -27 | -31 | -35 |
| **G** | **-8** | **1** | 2 | -2 | -6 | -10 | -14 | -18 | -14 | -18 | -22 | -26 |
| **A** | **-12** | -3 | 6 | 7 | 3 | -1 | -5 | -9 | -13 | -17 | -21 | -17 |
| **T** | **-16** | -7 | 2 | 3 | 12 | 8 | 4 | 0 | -4 | -8 | -12 | -16 |
| **C** | **-20** | -11 | -2 | -1 | 8 | 9 | 13 | 9 | 5 | 1 | -3 | -7 |
| **G** | **-24** | -15 | -6 | 5 | 4 | 5 | 9 | 10 | 14 | 10 | 6 | 2 |
| **A** | **-28** | -19 | -10 | -1 | 0 | 1 | 5 | 14 | 10 | 11 | 7 | **11** |

This gives us an alignment of

**A**

**|**

**A**

Note that the blue letters and gold arrows indicate the path leading to the maximum score.

We can continue to follow the path until we get to the following situation:

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **-** | **G** | **A** | **A** | **T** | **T** | **C** | **A** | **G** | **T** | **T** | **A** |
| **-** | **0** | **-4** | **-8** | **-12** | **-16** | **-20** | **-24** | **-28** | **-32** | **-36** | **-40** | **-44** |
| **G** | **-4** | **5** | 1 | -3 | -7 | -11 | -15 | -19 | -23 | -27 | -31 | -35 |
| **G** |  |  | **2** | -2 | -6 | -10 | -14 | -18 | -14 | -18 | -22 | -26 |
| **A** | **-12** | -3 | 6 | **7** | **3** | -1 | -5 | -9 | -13 | -17 | -21 | -17 |
| **T** | **-16** | -7 | 2 | 3 | 12 | **8** | 4 | 0 | -4 | -8 | -12 | -16 |
| **C** | **-20** | -11 | -2 | -1 | 8 | 9 | **13** | **9** | 5 | 1 | -3 | -7 |
| **G** | **-24** | -15 | -6 | 5 | 4 | 5 | 9 | 10 | **14** | **10** | **6** | 2 |
| **A** | **-28** | -19 | -10 | -1 | 0 | 1 | 5 | 14 | 10 | 11 | 7 | **11** |

The resulting global alignment is as follows:

**G A A T T C A G T T A**

**| | | | | |**

**G G A – T C – G - — A**

Remembering that the scoring scheme used was +5 for a match, -3 for a mismatch, and –4 for a gap, we can double check the score of the alignment:

**G A A T T C A G T T A**

**| | | | | |**

**G G A – T C – G - — A**

+ - + - + + - + - - +

5 3 5 4 5 5 4 5 4 4 5

5 – 3 + 5 – 4 + 5 + 5 – 4 + 5 – 4 – 4 + 5 = 11

so this alignment results in a global alignment score of 11.

Note that in the case of the sequence and scoring schemes we chose, there was only one maximal alignment. It is possible that there could be multiple alignments yielding the same score, as evidenced by having multiple ways to obtain the maximal score in a given cell in the scoring matrix. In such a case, the traceback can be accomplished in any manner desired, as long as the same set of rules is consistently used in order for reproducibility.