

CS 6670 Advanced Bioinformatics

Assignment 4

Gopal Menon

July 21, 2016

1. Calling Modes

The program can find the alignment score using both recursion and dynamic programming. If it is passed a parameter of Y , it will skip the recursion and do only the dynamic programming part. Otherwise it will do both. In the cases where long strings of maximum length upto 30 characters are generated, the recursive computation is skipped.

The main class that does the alignment can be used only in one of the two modes. If it is called in dynamic programming mode, it can be asked to print the alignment of the two strings.

2. Output format

The program generates two random nucleotide strings and then finds the alignment score or edit distance using dynamic programming. Then it uses the traceback algorithm to print out the string alignment.

The program generates the following output

- a) The two strings are printed out
- b) If applicable, the alignment string or edit distance found using the recursive mode is printed out
- c) The alignment string or edit distance found using the dynamic programming mode is printed out
- d) The alignment is then printed out in reverse order of the string. The following convention is used:
 - i. If a character in one string is substituted by another, then it is shown in the form $T \times A$, where T in the first string is replaced by an A .
 - ii. If a character in one string matches with a character in the other string, then this is shown using the form $C = C$.
 - iii. If a character needs to be inserted into the first string to match the second string, then it is shown using the form $_ C$.

3. Test Cases and Corresponding Output

Here are some instances of randomly generated strings and the output produced by the program.

First string: GCCGTA

Second string: TACCACGTGATCG

Alignment score by recursion: 8

Alignment score by Dynamic Programming: 8

```
-   G
-   C
-   T
A = A
-   G
T = T
G = G
-   C
-   A
C = C
C = C
-   A
G X T
```

First string: GAATCT

Second string: TCCATCG

Alignment score by recursion: 4

Alignment score by Dynamic Programming: 4

```
T X G
C = C
T = T
A = A
-   C
A X C
G X T
```

First string: CTACTGCTATTCCGC
Second string: TCGGACTCTTAAGCTTGG

Recursive alignment score computation skipped.

Alignment score by Dynamic Programming: 10

```
C X G
G = G
-   T
C X T
C = C
T X G
T X A
A = A
-   T
T = T
C = C
G   -
T = T
C = C
A = A
-   G
T X G
C = C
-   T
```