Alexa Buckner

Global Alignment with Affine Gap Penalty for TERT Comparison

Project:

Stress may cause critical shortening in telomeres, a buffer that protects the coding region of DNA. Work in humans has found that, in addition to shorter telomeres, chronically stressed caregivers of sick relatives have lower than average levels of telomerase, the enzyme responsible for elongating telomeres. The goal of this project was to create a program that mimicked the EMBOSS program used by Au et al. (2009) for comparing hTERT protein sequence, one component of human telomerase, to Japanese medaka (oTERT), pufferfish (fTERT), and zebrafish (zTERT). Au et al. compared several fish TERT proteins to hTERT to determine which would be a model organism for studying telomerase biology. I decided to perform the same task on the TERT protein sequence of *A. burtoni*, an African cichlid fish, because it's manipulate social structure makes it an interesting organism for studying the effects of stress on telomerase. If *A.* burtoni's TERT protein (burtoniTERT) is similar to hTERT, it may be useful for studying human-like telomerase biology.

The program I created calculates the percent identity of the aligned protein sequences (the number of matching amino acids divided by the total length of the alignment). In order to calculate identity, the program would find the best global alignment of two sequences with a 10-point penalty for opening a gap and a 0.5-point penalty for extending the gap. Global alignment and the specific penalties were chosen by first looking at EMBOSS's default parameters and comparing that to the data found by Au et al. for tuning my program.

Data:

All protein sequences were taken from the National Center for Biotechnology Information (NCBI) online database. I found the TERT protein sequences for humans (accession number NM_198253.2), medaka (*Oryzias latipes*-NM_001104816), zebrafish (*Danio reio*-NM_001083866), pufferfish (*Takifugu rubripes*-AY861384), and burtoniTERT (*Astatotilapia burtoni*-GBBJ01064744). I did not need to alter the sequences in any way.

Program:

First, I copied the protein sequences from a FASTA file directly into the program, they are referred to as string1 and string2.

There are three tables that are the length of one protein sequence (sequence1) and the width of the other (sequence2). They are populated with the best score at index i of string1 and index j of string2. One table (mid) corresponds to matches and mismatches between the strings. One table (top) corresponds to deletions in string1 and the last (bot) corresponds to insertions in string1. Each table has a corresponding backtrack table (backtrack, backtop, and backbot) to tell the computer how to best align the two strings. The BLOSUM62 scoring dictionary was used to determine the number of points added or subtracted for matching or mismatching amino acids. The dictionary is referred to as "scoredic".

The best possible scores were calculated as follows:

- Set mid[0][0] = 0, top[0][0] = -10000, and bot[0][0] = -10000
- Fill in first row of top table, then first col of bot table, then move row by row of all three tables (bottom table first, then top table, then middle table)

- For the top table: The score is the maximum of the score of top[i][j-1]- 0.5 and mid[i][j-1]-10. The backpack table for top was filled with "top" or "mid" depending on if the top score or mid score was higher.
- For the bottom table: The score is the maximum of the score of bot[i][j-1]- 0.5 and mid[i][j-1]-10. The backtrack table for bot was filled with "bot" or "mid" depending on if the bottom score or mid score was higher.
- For the middle table: The score is the maximum of the score of top[i][j], bot[i][j], and mid[i-1][j-1]-scoredic[string1[i-1]][string2[j-1]].
    - If the maximum score was found by (mis)matching amino acids, backtrack[i][j] = "dia" for diagonal movement.
    - If the max score was closing a deletion or insertion, backtrack[i][j] = "top" or "bot".
- The backtrack tables were used to find the alignment of the two strings by starting a while loop in the bottom left node of backtrack and moving until it reached the i = 0 and j = 0. Two empty alignment strings, alignment1 and alignment2, were created.
- If backtrack[i][j] = "top" we move to the top backtrack table and make a gap in alignment1 and alignment2 = string2[j-1]. We then move left (j-1) and check if the top backtrack table still says we have a deletion. If it doesn't say "top" we exit the if loop and move back to the middle backtrack table. The same process happens for the bottom.
- If the middle backtrack table says "dia", we know the alignment is a (mis)match and we add a letter from both strings to the alignment strings. We then move diagonally up (i-1 and j-1).
- To calculate identity, we see if the character i in alignment1 matches character i in alignment2. If they match, we add 1 to the match score. Identity is the match score/length of the alignment*100.

Results:

The initial length of burtoniTERT shows us that it is about the same length as the TERT proteins from the other fish, which doesn't prove that the its identity with hTERT will be significantly different. The global alignment program I wrote gives me a very similar score and identity calculated by Au et al. with EMBOSS, but the scores, number of matched amino acids, and total length of the aligned sequences are not exactly the same. Since I am using the same gap open, gap extension, and BLOSUM scores, I expect that the differences in the scores is due to different ordering of the if statements. I calculated the score of node [i][j] in the bottom table first, then the top table, then in the middle table. EMBOSS could have done this differently. As I do not know how EMBOSS chooses between ties, I cannot make my program exactly identical. The fact that the percent identities I calculated are the same as the percent identities calculated by Au et al., I know that my program is sufficient for comparing hTERT to burtoniTERT.

Table 1: The length of the TERT protein, percent identity to hTERT calculated by EMBOSS and my program, and the score of the alignments of fish TERT and hTERT.

	hTERT	oTERT	zTERT	fTERT	burtoniTERT
Protein (aa)	1132	1090	1098	1074	1099
%identity	--	431/1169 = 37	395/1178 = 34	409/1158 = 35	411/1167 = 35
%identity w/ hTERT calc by EMBOSS	--	435/1189 = 37	403/1188 = 34	418/1197 = 35	413/1193 = 35
My score from global alignment (emboss score)	5961 (5961)	1708 (1706)	1493.5 (1496)	1620.5(1617)	1629.5 (1627.5)

The percent identity EMBOSS calculated for burtoniTERT and hTERT is 35%, the same identity I calculated. This calculated identity is the same as the identity between the pufferfish and human. It is 1% more identical to hTERT than zebrafish. It is also 2% worse than the identity between medaka TERT and human TERT. This suggests that *A. burtoni* is not a better organism for studying telomerase biology than medaka. This is surprising because *A. burtoni* and the Japanese medaka are close relatives, so I expected them to have a closer identity. This does not disqualify it from being a good organism for studying telomerase biology in relation to stress, as *A. burtoni* have a well-documented social structure that can be easily manipulated. It does mean, however, that any results from *A. burtoni* research should be carefully scrutinized before extrapolating to human telomerase. The program I created can be used to test the identity between any peptide sequences, and can be used to compare more organisms TERT sequence to hTERT to find model for studying stress and telomerase more identical to human TERT.

References:

Au, D.W.T., Mok, H.O.L., Elmore, L.W., and Holt, S.E. (2009). Japanese medaka: A new vertebrate model for studying telomere and telomerase biology. Comp. Biochem. Physiol. Part C Toxicol. Pharmacol. *149*, 161–167.