Solution Set 1

Due 11:59pm, Saturday, September 16th

Collaboration is allowed on this homework. You may discuss the problems with your colleagues, but each student must prepare and submit a separate assignment. Please list the names of the people you worked with:

The goal of this assignment is to help you to understand the details of how pairwise alignment algorithms work. To achieve this, you should work out the alignments by hand. Do not use an alignment program to calculate the alignments on this problem set. Please provide answers to all questions. In order to obtain full credit, explain your reasoning on each question and show all intermediate steps leading to your solution.

You may submit your assignments in any of the following ways:

- Download the assignment in pdf format and print it out. Write your solutions in the space provided. Scan your handwritten solution and upload it to Canvas. Attach additional pages, if needed.
- Download the assignment in pdf format. Use the commenting features in adobe or a similar tool to enter your solution directly on the pdf. Upload the completed assignment, annotated with your solutions.
- Download the assignment in latex format. Enter your solutions in latex format, compile the assignment, and turn in the resulting pdf.

In Problems 1 and 2, you are asked to align pairs of sequences. An alignment template, consisting of a grid with cells for the alignment score and the traceback arrows, is provided for these problems. Submit the alignment matrices with your assignment in one of the following ways:

- Download alignmentTemplate.pdf. Using a pad or tablet, fill in the template with a stylus. Save the annotated pdf and upload it to Canvas with your assignment.
- Download and print out alignmentTemplate.pdf. Enter the alignment scores and traceback arrows in the appropriate cells with a pen or pencil. Scan your handwritten alignment matrix and upload it to Canvas.
- Download alignmentTemplate.xlsx. Using excel as a tool for recording a table, type the alignment scores and traceback arrows in the appropriate cells of the template. Save the result as a pdf and upload it to Canvas.

1. **Global pairwise alignment:** This assignment asks you to align s_1 = HOUSTON and s_2 = TUCSON, using a scoring scheme that assigns a value of 2 to matches, a value of -1 to mismatches, and a value of -1 to gaps.

   (a) Calculate the global alignment of s_1 and s_2 using this scoring scheme. Hand in your alignment matrix with scores and traceback on the alignment template provided.

	0		H		O		U		S		T		O		N
0	0	--	-1		-2		-3		-4		-5		-6		-7
		\		\		\		\		\					
T	-1		-1	--	-2	--	-3	--	-4		-2	--	-3	--	-4
		\	|	\		\						\		\	
U	-2		-2		-2		0	--	-1	--	-2	--	-3	--	-4
		\	|	\	|		|	\		\		\		\	
C	-3		-3		-3		-1		-1	--	-2	--	-3	--	-4
		\	|	\	|		|	\							
S	-4		-4		-4		-2		1	--	0	--	-1	--	-2
		\	|	\				|	\		\				
O	-5		-5		-2	--	-3		0		0		2	--	1
		\	|		|	\			|	\	|		|	\	
N	-6		-6		-3		-3		-1		-1		1		4

Two optimal alignments:

		H	O	U	-	S	T	O	N
Score	4	-1	-1	2	-1	2	-1	2	2
		-	T	U	C	S	-	O	N

		H	O	U	-	S	T	O	N
Score	4	-1	-1	2	-1	2	-1	2	2
		T	-	U	C	S	-	O	N

   (b) How many different optimal alignments are there? Show them.

```
HOU_STON
_TUCS_ON

HOU_STON
T_UCS_ON
```

   Both alignments have a score of 4.

2. **Local alignment:**

   (a) Compute the local alignment of s_1 = `HABAKUK` and s_2 = `HAIKU`, using the following scoring system: matches = 2, mismatches = -1, indels = -2. Hand in your alignment matrix with scores and traceback on the alignment template provided.

	0	H	A	B	A	K	U	K
0	0	0	0	0	0	0	0	0
H	0	2	0	0	0	0	0	0
A	0	0	4	2	2	0	0	0
I	0	0	2	3	1	1	0	0
K	0	0	0	1	2	3	1	2
U	0	0	0	0	0	1	5	3

   (b) What is the optimal local alignment score? Show all optimal local alignment(s).

```
HABAKU
HAI_KU

HABAKU
HA_IKU
```

   Both alignments have a score of 5.

3. **Scoring functions for local alignments:**

   (a) What four properties must a scoring function satisfy to be appropriate for local alignment?

   - *Local alignments can be found by maximizing a simiaritly score, only. This is contrast to global alignment, where the opimization function can a similarity function or a distance function.*
   - *The expected alignment score in unrelated sequences must be less than zero. The expected alignment score depends on match and mismatch scores and the background frequencies of symbols in the alphabet.*
   - *The scoring function must have a positive value for at least one pair of symbols, $(x, y), x, y \in \Sigma$. For the simple scoring function used in this assigment, the score is positive for all pairs, $(x, x), x \in Sigma$, because $M > 0$. Note that this is a property of the scoring function, not a property of a specific alignment. The expected alignment score depends on the match and mismatch scores, but not the gap penalty. We do not have a model of the background fequency of gaps.*
   - *We require $m \geq g$. Otherwise, gaps would always be preferred over substitutions.*

(b) Verify that the scoring function that you used in Problem 2 satisfies these criteria.

- *A similarity score is a maximization function. The values of M, m, and g satisfy the criteria for a similarity score:*

$$M > 0; m, g < 0$$

- *The expected alignment score should be negative. Since we are asked to align English words in this problem, the expected alignment frequency can be calculated using letter frequencies in English texts, as follows:*

$$\begin{aligned}\bar{S} &= \sum_{x\in\Sigma} p_x^2 \cdot M + \sum_{x\in\Sigma} \sum_{\substack{y\in\Sigma,\\ y\neq x}} p_x p_y \cdot m \\ &= 0.0657 \cdot M + 0.937 \cdot m \\ &= -0.806\end{aligned}$$

  For $M = 2, m = -1$, and typical letter frequencies in the English language, the alignment score is less than 0.
  Letter frequencies retrieved from https://en.wikipedia.org/wiki/Letter_frequency. Assume Σ consists of the all 26 letters in the English alphabet.
- *The scoring scheme has a positive entry ($M = 2$), for all 26 pairs $(x, x), x \in \Sigma,$. Here, Σ consists of the letters in the English alphabet.*
- $m = -1; 2g = -4$, *so* $m > 2g$

4. **Scoring functions for local alignments:**

   (a) When aligning `HABAKUK` and `HAIKU`, It is possible to specify a scoring function (i.e., to assign values to M, m, and g) for which

      i. the following alignment is optimal

```
HA
HA
```

      ii. the alignment(s) you obtained in Problem 2(a) is(are) sub-optimal.

   Give an example of such a scoring function. Your scoring function should satisfy the criteria in Problem 3. What is the score of the above alignment (`HA-HA`) with this scoring function?

   $M = 2, m = -3, g = -2$ is one example.

   With these values, the score of the alignment in above is $2M = 4$.

   (b) Rescore the alignments you obtained in Problem 2 with this scoring function. What score do you obtain? (Note you do not need to work another alignment matrix to answer this question.)

```
H A B A K U
H A I _ K U
2 2 -3 -2 2 2
```

```
H A B A K U
H A _ I K U
2 2 -2 -3 2 2
```

   When $M = 2, m = -3, g = -2$, the alignments in Problem 2 have a score of 3.

(c) In fact, there are an infinite number of scoring functions for which the `HA-HA` alignment has a higher score than the alignment(s) you obtained in Problem 2(a). Give an equation or inequality in terms of M, m, and g that specifies this set of scoring functions. It is not necessary to restate the criteria in Problem 3; simply give the additional constraints required. Explain your reasoning. (Note you do not need to work another alignment matrix to answer this question.)

Aligning `HA-HA` results in 2 matches which is a score of $2M$*. The alignment in Problem 2 results in 4 matches, 1 mismatch and 1 gap, which is a score of* $4M + m + g$*. A scoring function that prefers `HA-HA` must satisfy the inequality*

$$4M + m + g < 2M$$

$$2M + m + g < 0$$

(d) Is there a scoring function for which the `HA-HA` alignment is a *unique* optimal alignment? If so, given an example of such a scoring function. If not, why not?

No. The "`KU`" in`HABAKUK` and "`KU`" in`HAIKU` form an alignment of two matches that does not overlap with the `HA-HA` alignment. Given simple scoring function with a single match score for all symbols, these two alignments will always have the same score. Either both are optimal or neither one is optimal.

```
KU
KU
```

(e) Is there a scoring function for which the following alignment is an optimal local alignment? If so, given an example of such a scoring function. If not, why not?

```
H A B A K
H A _ I K
```

No. Any alignment that contains `HABAK` and `HA_IK` can be extend to include the `U`s in `HABAK` and `HAIKU`, which will increase the score

It is not possible to construct a scoring scheme for which $3M + g + m > 4M + g + m$.