### Profile HMM with Pseudocounts Problem

*Construct a profile HMM with pseudocounts from a multiple alignment.*

**Input:** A multiple alignment *Alignment*, a threshold *θ*, and a pseudocount *σ*.

**Output:** HMM(*Alignment*, *θ*, *σ*), in the form of transition and emission matrices.

**Input Format.** The first line of the input contains the threshold *θ* and the pseudocount *σ* (space-delimited). The second line of the input is “--------” (a delimiter). The third line of the input is the list of symbols in the alphabet *Σ* (space-separated). The fourth line of the input is “--------” (a delimiter). The remaining lines are the multiple alignment *Alignment*, with one sequence (containing gaps) on each line.

**Output Format.** Let *States* denote the set of all states in the resulting profile HMM, where we have one Match state for each column of *Alignment* with fewer than *θ* gaps (M1, M2, …), one Insertion state for every Match state as well as before the first Match state (I0, I1, I2, …), one Deletion state for every Match state (D1, D2, …), a Start state (S), and an End state (E). The first |*States*|+1 lines should contain the transition matrix of HMM(*Alignment*, *θ*), the next line should be “--------” (a delimiter), and the remaining |*States*|+1 lines should contain the emission matrix of HMM(*Alignment*, *θ*, *σ*), with both matrices drawn in the format as shown in the sample output.

**Constraints.** 0.2 ≤ *θ* ≤ 0.4; *σ* = 0.01 ; |*Σ*| = 5; 5 ≤ |*Alignment*| ≤ 10; 80 ≤ |*Alignment*[*i*]| ≤ 100

**SAMPLE DATASET:**

Input:

0.358 0.01

--------

A B C D E

--------

A-A

ADA

ACA

A-C

-EA

D-A

Output:

S I0 M1 D1 I1 M2 D2 I2 E

S 0 0.01 0.819 0.172 0 0 0 0 0

I0 0 0.333 0.333 0.333 0 0 0 0 0

M1 0 0 0 0 0.398 0.592 0.01 0 0

D1 0 0 0 0 0.981 0.01 0.01 0 0

I1 0 0 0 0 0.01 0.981 0.01 0 0

M2 0 0 0 0 0 0 0 0.01 0.99

D2 0 0 0 0 0 0 0 0.5 0.5

I2 0 0 0 0 0 0 0 0.5 0.5

E 0 0 0 0 0 0 0 0 0

--------

A B C D E

S 0 0 0 0 0

I0 0.2 0.2 0.2 0.2 0.2

M1 0.771 0.01 0.01 0.2 0.01

D1 0 0 0 0 0

I1 0.01 0.01 0.327 0.327 0.327

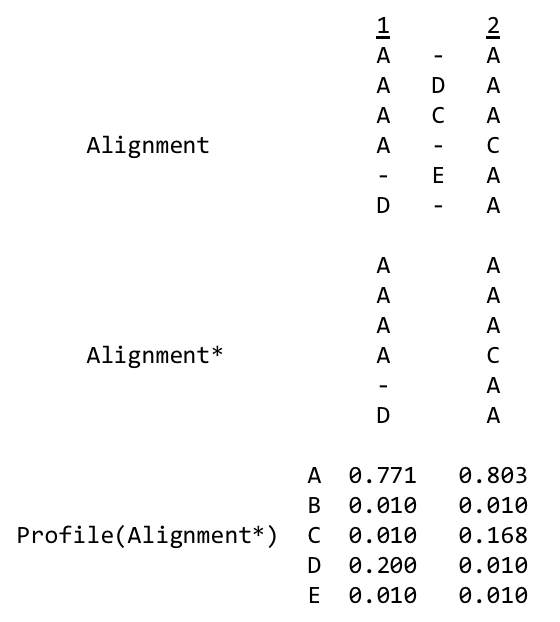
M2 0.803 0.01 0.168 0.01 0.01

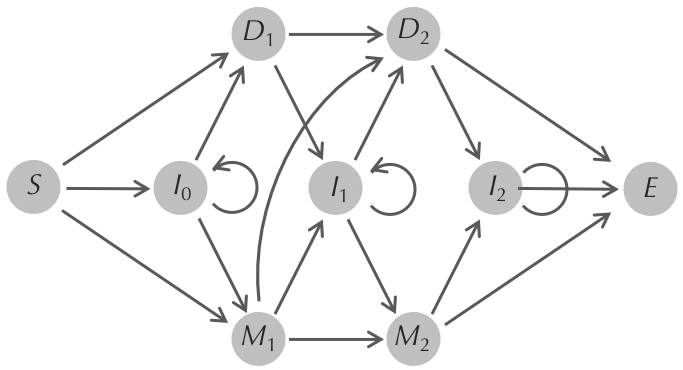
D2 0 0 0 0 0

I2 0.2 0.2 0.2 0.2 0.2

E 0 0 0 0 0

The above transition and emission matrices are the result of constructing a profile-HMM from the given multiple alignment *Alignment* using a threshold of *θ* = 0.289 and a pseudocount of *σ* = 0.01.





**TEST DATASET 1:**

Input:

0.1 0.01

--------

A

--------

A

Output:

S I0 M1 D1 I1 E

S 0 0.00971 0.981 0.00971 0 0

I0 0 0.333 0.333 0.333 0 0

M1 0 0 0 0 0.00980 0.990

D1 0 0 0 0 0.5 0.5

I1 0 0 0 0 0.5 0.5

E 0 0 0 0 0 0

--------

A

S 0

I0 1.0

M1 1.0

D1 0

I1 1.0

E 0

This dataset makes sure that your code is correctly applying the pseudocounts to the transition matrix. Make sure that your implementation of profile HMMs without pseudocounts passes all test cases before doing this problem. This dataset only has one character in its multiple alignment, so only the only observed transitions were from state S to state M1 to state E. All other transitions that are possible given the design of the profile HMM are not observed. If your output does not match the correct output your code is likely incorrectly applying the pseudocounts. Make sure that you are adding the specified pseudocount to the transition and emission **probabilities**, not the counts. A common mistake is to add pseudocounts to the profile created for the transition and emission matrices and then normalize the result. This how pseudocounts were used in Motif Finding. Make sure that your code is first finding the emission and transition matrices of the basic profile HMM and then adding pseudocounts to those probabilities and normalizing. Also make sure that all unseen but possible transitions are accounted for. In this dataset states I0, D1, and I1 are never visited but have transition probabilities because of the use of pseudocounts.

**TEST DATASET 2:**

Input:

0.1 0.01

--------

A B

--------

A

Output:

S I0 M1 D1 I1 E

S 0 0.00971 0.981 0.00971 0 0

I0 0 0.333 0.333 0.333 0 0

M1 0 0 0 0 0.00980 0.990

D1 0 0 0 0 0.5 0.5

I1 0 0 0 0 0.5 0.5

E 0 0 0 0 0 0

--------

A B

S 0 0

I0 0.5 0.5

M1 0.990 0.00980

D1 0 0

I1 0.5 0.5

E 0 0

This dataset makes sure that your code is correctly applying the pseudocounts to the emission matrix. This dataset is identical to Test Dataset 1 except that there is one more character in the alphabet: B. The emission matrix should include a column for the emission probabilities of character B. Without pseudocounts our emission matrix for this profile HMM would have 0 probability of emitting B. Once pseudocounts are added the emission probabilities for the non-silent states change. If your output does not match the correct output make sure that you are applying pseudocounts to the emission matrix. Remember that pseudocounts are added to the **probabilities**, not the profile counts. Also be sure that non-silent states that were never seen in the multiple alignment (I0, I1) have non-zero emission probabilities. States like these should have equal probabilities of emitting any character.

**TEST DATASET 3:**

Input:

0.1 0.5

--------

A B

--------

A

Output:

S I0 M1 D1 I1 E

S 0 0.2 0.6 0.2 0 0

I0 0 0.333 0.333 0.333 0 0

M1 0 0 0 0 0.25 0.75

D1 0 0 0 0 0.5 0.5

I1 0 0 0 0 0.5 0.5

E 0 0 0 0 0 0

--------

A B

S 0 0

I0 0.5 0.5

M1 0.75 0.25

D1 0 0

I1 0.5 0.5

E 0 0

This dataset makes sure that your code is correctly applying the correct pseudocount. This dataset is identical to Test Dataset 2 except for the value of the pseudocount. If your code passed Test Dataset 2 but does not have the correct output for this dataset you are likely not using the pseudocount value from the input and are adding some constant of your own choice. It is also possible that your normalization after applying the pseudocounts was not correctly implemented and relied on the pseudocount being of some value.

**TEST DATASET 4:**

Input:

0.4 0.01

--------

A B

--------

AB

A-

Output:

S I0 M1 D1 I1 E

S 0 0.00971 0.981 0.00971 0 0

I0 0 0.333 0.333 0.333 0 0

M1 0 0 0 0 0.5 0.5

D1 0 0 0 0 0.5 0.5

I1 0 0 0 0 0.00980 0.990

E 0 0 0 0 0 0

--------

A B

S 0 0

I0 0.5 0.5

M1 0.990 0.00980

D1 0 0

I1 0.00980 0.990

E 0 0

This dataset makes sure that your code is correctly applying the threshold when determining the seed alignment. It is analogous to Test Dataset 3 in the Profile HMM problem. Since half of the second column is gap characters the second column will not be a part of the seed alignment (0.5 > threshold of 0.4). If your output has the wrong states your implementation of the regular profile HMM is likely incorrect. If your implementation passed all the tests for the regular profile HMM problem but fails this test then be sure that your code does not invalidate that implementation when adding pseudocounts **TEST DATASET 5:**

Input:

0.4 0.01

--------

A B

--------

A-

-A

-B

Output:

S I0 M1 D1 I1 E

S 0 0.333 0.657 0.00971 0 0

I0 0 0.00971 0.00971 0.981 0 0

M1 0 0 0 0 0.00980 0.990

D1 0 0 0 0 0.00980 0.990

I1 0 0 0 0 0.5 0.5

E 0 0 0 0 0 0

--------

A B

S 0 0

I0 0.990 0.00980

M1 0.5 0.5

D1 0 0

I1 0.5 0.5

E 0 0

This dataset makes sure that your code correctly handles deletion states. It is analogous to Test Dataset 4 in the Profile HMM problem. If your output has the wrong states your implementation of the regular profile HMM is likely incorrect. If your implementation passed all the tests for the regular profile HMM problem but fails this test then be sure that your code does not invalidate that implementation when adding pseudocounts.

**TEST DATASET 6:**

Input:

0.5 0.01

--------

A B

--------

AA-

--A

--B

Output:

S I0 M1 D1 I1 E

S 0 0.333 0.657 0.00971 0 0

I0 0 0.495 0.00971 0.495 0 0

M1 0 0 0 0 0.00980 0.990

D1 0 0 0 0 0.00980 0.990

I1 0 0 0 0 0.5 0.5

E 0 0 0 0 0 0

--------

A B

S 0 0

I0 0.990 0.00980

M1 0.5 0.5

D1 0 0

I1 0.5 0.5

E 0 0

This dataset makes sure that your code allows for insertion states to transition to themselves. It is analogous to Test Dataset 5 in the Profile HMM problem. If your output has the wrong states your implementation of the regular profile HMM is likely incorrect. If your implementation passed all the tests for the regular profile HMM problem but fails this test then be sure that your code does not invalidate that implementation when adding pseudocounts.

**TEST DATASET 7:**

Input:

0.9 0.01

--------

A B

--------

A-

-A

Output:

S I0 M1 D1 I1 M2 D2 I2 E

S 0 0.00971 0.495 0.495 0 0 0 0 0

I0 0 0.333 0.333 0.333 0 0 0 0 0

M1 0 0 0 0 0.00971 0.00971 0.981 0 0

D1 0 0 0 0 0.00971 0.981 0.00971 0 0

I1 0 0 0 0 0.333 0.333 0.333 0 0

M2 0 0 0 0 0 0 0 0.00980 0.990

D2 0 0 0 0 0 0 0 0.00980 0.990

I2 0 0 0 0 0 0 0 0.5 0.5

E 0 0 0 0 0 0 0 0 0

--------

A B

S 0 0

I0 0.5 0.5

M1 0.990 0.00980

D1 0 0

I1 0.5 0.5

M2 0.990 0.00980

D2 0 0

I2 0.5 0.5

E 0 0

This dataset makes sure that your code includes all relevant columns in the seed alignment. It is analogous to Test Dataset 6 in the Profile HMM problem. Since the threshold value in this dataset is so high, all columns in the original multiple alignment should be included in the seed alignment. If your output has the wrong states your implementation of the regular profile HMM is likely incorrect. If your implementation passed all the tests for the regular profile HMM problem but fails this test then be sure that your code does not invalidate that implementation when adding pseudocounts.