Introduction to Bioinformatics
Making and using Position-Specific Scoring Matrices (Part II)

You now know how to make a PSSM based on an alignment of protein-binding sites (see notes entitled *Using PSSMs*). You know how to make the matrix with or without pseudocounts. You can make it by hand or using the convenient function (MAKE-PSSM-FROM …). Of course, the matrix doesn't do you any good until you apply it to sequences that may or may not be protein-binding sites, e.g. to an entire genome.

In principle, you already know how to do this too (see notes entitled *Position-specific scoring matrices*). Scoring a candidate sequence is simple (Table 2 from the aforementioned notes): (1) apply the PSSM to each position of the sequence, multiplying together each of the fractions you get., (2) apply the nucleotide background frequencies to each position of the sequence, multiplying together each of the fractions you get, and (3) divide the first product by the second. This procedure gives you the degree to which the PSSM is better at predicting the candidate sequence than the background nucleotide frequencies. If the ratio is greater than 1, then the PSSM is better, while if the ratio is less than 1, the background frequencies are the better predictor.

A. Using scoring matrices to score individual sequences

Heh, heh, heh… Let's do it. Take the first sequence in S6803-ntcA-sites as an example. If you don't have S6803-ntcA-sites in your space already, then create it as described in the notes entitled *Making and Using Position-Specific Scoring Matrices (Part I)*.

SQ1. Write a loop that goes through each letter of the sequence and, simultaneously, each column of `freq-table`. You might follow this template:

```
(FOR-EACH column FROM fill in
     INITIALIZE raw-score = 1
     FOR letter IN S6803-ntcA-sites[1][2]
     AS s = fill in
     (ASSIGN raw-score = fill in
     FINALLY (RETURN raw-score))
```

Hint: Don't try to do the entire loop at once. Instead, display key variables as you go, to make sure that things are as you expect. For example, here's one intermediate in building the loop:

```
(FOR-EACH column FROM fill in
     INITIALIZE score = 1
     FOR letter IN S6803-ntcA-sites[1][2]
     (DISPLAY-LINE column letter))
```

SQ1a. You may get the following error:

```
While executing the IMPLIED-DO clause "((ASSIGN RAW-SCORE = ... "
After executing the loop 40 times, an error was detected:
`NIL' is not of the expected type `NUMBER'
```

What happened the 40th time through the loop? How can you fix the problem? A work-around (not a real fix) is to proceed through the loop only for the first 39 columns.

SQ2. Define a variable `seq` that is identical to the first sequence in `S6803-ntcA-sites` except that the first letter is a G. (Hint: First define the variable to be *identical* to the first sequence in `S6803-ntcA-sites`, and then change the first letter to a G.) Modify your loop from SQ1 to calculate a new raw score, based on `seq`. How does this raw score compare to the one calculated in SQ1?

SQ3. Suppose one of your colleagues (maybe even you!) obtained a raw score of zero in SQ2. How would you explain it?

SQ4. What does the raw score mean mechanistically (i.e. how can you explain how it was obtained)? What does it say regarding the relationship between `seq` and proven NtcA-binding sites?

This raw score doesn't say much. It is a very small number, but small compared to what? The second part of the scoring strategy is to compare the number to that obtained from using a different matrix, one in which each cell is the probability of finding a specific nucleotide if the genome were mashed up. You can obtain the background frequencies of the nucleotides by counting each nucleotide and dividing each of the four counts by the total number of nucleotides… but you've done that before. Let me save some sweat by introducing another convenient function:

```
(BACKGROUND-FREQUENCIES-OF fill-in)
```

SQ5. What are the background nucleotide frequencies in the genome of *Synechocystis* PCC 6803?

SQ6. What are the background nucleotide frequencies in just the intergenic sequences of *Synechocystis* PCC 6803? How do these frequencies compare with those derived from the entire genome?

SQ7. Rewrite the loop from SQ3, except using the background nucleotide frequencies as the source of frequencies rather than `freq-table`. This may sound like a simple change, but because `freq-table` is a table and you have the background frequencies as a list, there are Complications. Avoid them by availing yourself of the following function:

`(TO-TABLE-FROM` *list* `WITH-INDICES` *list-of-indices*`)`

SQ8. Take the ratio of the two raw-scores (from SQ3 and SQ7). In moderately plain English, what can you conclude?

B. Using scoring matrices to score a genome-worth of sequences

So you really can score a sequence using a PSSM and the background frequencies. That's OK for one sequence. How do we apply the procedure to an entire genome? Again, the principle is simple:

1. Starting at the beginning of the genome, extract a sequence the size of the PSSM (call this the ***window***).
2. Score the window.
3. If the score exceeds a given threshold, keep it, otherwise throw it away.
4. Move the window over by one nucleotide
5. Repeat steps 2-4 until you've gotten to the end of the genome.

You should be able (with some struggle, I admit) be able to do all of these steps within BioBIKE. However, rejoice! We have done it for you, because (a) the code you would be able to imagine would be too inefficient to go through an entire genome in a reasonable length of time (not your fault – it's just the nature of the language), and (b) lots of people want to scan genomes. The function that does the job has the syntax:

`(APPLY-PSSM-TO` *sequence* `WITH-PSSM-FROM` *aligned-sequences*`)`

`(APPLY-PSSM-TO` *set-of-sequences* `WITH-PSSM-FROM` *aligned-sequences*`)`

SQ9. Cut down `S6803-ntcA-sites` to extract just the first 39 nucleotides (the nucleotides prior to the gaps). Give it a name and use it in subsequent operations.

SQ10. Use APPLY-PSSM-TO and the aligned sequences you made in SQ9 to look for other NtcA-binding sites in the intergenic sequences of S6803. You may find good use for:

`(INTERGENIC-SEQUENCES-OF` *organism* `LABELED)`