Introduction to Bioinformatics
Search for Regulatory Sequences

I. The problem: How to find unknown regulatory sequences

In the recent notes entitled Gene regulation and bacteriophage, a great deal of attention was lavished on only about 100 nucleotides. Those 100 nucleotides are important, however. They contain the sequences that determine whether phage lambda will kill its host or instead integrate itself into the host genome. Most importantly, they do so not by virtue of encoding a key protein but by their ability to ***bind*** proteins. It is the specific sequence that is important, not what it encodes.

In a similar fashion, noncoding DNA regulates the expression of genes not just in phages but in all cellular life. Since our goal this semester is to find hitherto unrecognized DNA sequences that are of potential biological importance, we would do well to attend to sequences that are of undisputed importance – regulatory sequences – and learn how to recognize them.

Of course with lambda we cheated. The critical regulatory region upstream from the *cI* gene has long been known. Would we be able to find it if we didn't know where to look? Lambda provides us with a test, a known case with which to try out our proposed methods. You'll recall that in the same way you tested your methods to find origins of replication using the known case of the origin of replication of E. coli. Testing methods with a known case is a good general strategy.

Examine the proven lambda Repressor and Cro binding sites (operators) shown at the right (taken from Figure 5 of the previous notes), and consider the methods you used to find origins of replication.

Operators in phage lambda

```
OR1 TATCACCGCCAGAGGTA
OR2 TAACACCGTGCGTGTTG
OR3 TATCACCGCAAGGGATA
```

SQ1. Would you be able to find these operators by looking for a cluster of exact matches in the sequence of phage lambda? Why (not)? Try it!

SQ2. Would you be able to find these operators by looking for a cluster of matches to a pattern? Why (not)? Try it!

Again, it's no great honor to find regulatory sequences in lambda. Lambda is just a test case. The method should be considered good only if it finds regulatory sequences in other phages as well. With that in mind, looking for a pattern ***might*** work, if the pattern is well conserved amongst phages. It may be or it may not be. No way to tell but to look. More generally, however, any phage that has a repressor has to regulate its expression, and so the operon containing the repressor would certainly be expected to have regulatory sequences that control its expression.

We might be able to find these sequences in an unknown phage genome even if there is no conserved sequence amongst phages or even a conserved pattern. The trick is to look for a sequence – any sequence -- that is repeated multiple times, allowing for some degree of mismatches (just as the lambda operators have mismatches in their sequences). This sounds like a job for COUNT-OF-K-MERS?

SQ3. Would COUNTS-OF-K-MERS be able to find the lambda operators shown above? Why (not)? Try it!

Not exact matches, not patterns,… then how to look for these things? Have you done something like this before? …Yes! Each position of the operators have certain ***tendencies,*** biases towards

certain nucleotides, even if there is no absolute requirement. You've determined nucleotide tendencies before. Recall the use of PSSMs (Position Specific Scoring Matrices) for this purpose in What is a Gene, Part III? [1] There you provided sequences (upstream sequences of all coding genes of a genome) and constructed a PSSM. Unfortunately, the table (or graph) of tendencies you got wasn't enough to allow you to pick out a specific sequences of interest. The BioBIKE function MOTIFS-IN[2] takes PSSMs a step further. It searches for sequences within a set that allows you to create the ***best*** PSSM, i.e. a PSSM in which there is the least variability amongst the nucleotides in a column. So, in the end you get from MOTIFS-IN a set of nucleotide tendencies that are the most divergent from what you would expect from chance, i.e. from a random collection of short sequences.

II. MOTIFS-IN: How it finds a subset of sequences with minimum variability

Here's an illustration of how MOTIFS-IN works. Suppose you have upstream sequences from several eukaryotic genes (shown at the right) and have reason to believe that they contain binding sites for the same protein. You can look for the protein's binding sites in the following way:

Gene	Upstream sequence
snRNA U1 (pU1-6)	AGGTATATGGAGCTGTGACAGGGCAGAAGTGTGTGAAGTC
histone H1t	GCCCTACCCTATATAAGGCCCCGAGGCCGCCCGGGTGTTT
HMG-14	CGGCCGGCGGGGAGGGGGAGCCCGCGGCCGGGGACGCGGG
TP1	GCCAAGGCCTTAAATACCCAGACTCCTGCCCCCGGGCCTT
protamine P1	CCCTGGCATCTATAACAGGCCGCAGAGCTGGCCCCTGACT

Sequences after completion of step 2. The arbitrarily chosen candidate motif is highlighted in red and the best matches in the other sequences are highlighted in pink.

Step 1: Arbitrarily choose a candidate motif from one of the sequences

Step 2: Find the best match to the candidate motif in all the other sequences

Step 3: Construct a PSSM based on an alignment of the matches

Alignment (first)

```
ACAGGGCAGAA
CCCGGGTGTTT
CCGGGGACGCG
CCCCCGGGCCT
CCGCAGAGCTG
```

Position-Specific Scoring Matrix (PSSM)

	1	2	3	4	5	6	7	8	9	10	11
A	0.2	0.0	0.2	0.0	0.2	0.0	0.4	0.2	0.0	0.2	0.2
C	0.8	1.0	0.4	0.4	0.2	0.0	0.2	0.2	0.4	0.4	0.0
G	0.0	0.0	0.4	0.6	0.6	1.0	0.2	0.6	0.4	0.0	0.4
T	0.0	0.0	0.0	0.0	0.0	0.0	0.2	0.0	0.2	0.4	0.4

SQ4. What is the relationship between the alignment of the best matches (shown above on the left) with the PSSM of that alignment (shown above on the right?

Step 4: Calculate how good this PSSM is, by comparing the probability of each sequence in the alignment using the PSSM vs using overall nucleotide frequencies

Step 5: Extend the matches if it increases the efficacy of the PSSM

Step 6: Save the score and go back to Step 1 and do it again

The alignments that produce the best PSSMs are returned as the result, and these alignments give you an idea of what variability is tolerated in the proposed motif. You get as many alignments as you request (or 3 if you accept the default value. You also get an E-value for each alignment.

SQ5. What do you think the E-value means (use its meaning in Blast as inspiration)?

[1] If you don't recall what a PSSM is, then revisit that exercise.
[2] MOTIFS-IN is based on the program called MEME, whose workings are loosely described here.

SQ6. MOTIFS-IN does not guarantee that it will always return the best motif. You might get a different motif if you run the function a second time. Why?

Here's the best alignment found by MOTIFS-IN, given the sequences shown above:

Alignment (final)

```
TATATAAGGCC
TATATGGAGCT
TATAACAGGCC
CTTAAATACCC
GAGGGGGAGCC
```

Position-Specific Scoring Matrix (PSSM)

	1	2	3	4	5	6	7	8	9	10	11
A	0.0	0.8	0.0	0.8	0.4	0.4	0.4	0.6	0.0	0.0	0.0
C	0.2	0.0	0.0	0.0	0.0	0.2	0.0	0.0	0.2	1.0	0.8
G	0.2	0.0	0.2	0.2	0.2	0.4	0.4	0.4	0.8	0.0	0.0
T	0.6	0.2	0.8	0.0	0.4	0.0	0.2	0.0	0.0	0.0	0.2

SQ7. Does the PSSM of this alignment look better (less variability) than the first? Why (not)?

SQ8. MOTIFS-IN thought this alignment was considerably better than the first. Reread the criteria in Step 4 carefully and reconsider your answer to SQ7.

SQ9. From the PSSM, derive a consensus sequence, composed of the most common nucleotide at each position. If there is no most common nucleotide, use a place-holder (e.g. "n" or "."). What motif do you think you found? Bear in mind that I said that these were upstream sequences from eukaryotic genes.

If this method is good, you'd think it should be able to find the repressor-binding sites of lambda. Let's give it a shot. However, you will greatly improve your chances by decreasing the target size as much as possible, otherwise MOTIFS-OF is prone to find matches that have nothing to do with regulatory sequences (e.g. commonly occurring codons that lie next to each other). Here's some advice:

- Limit the set of sequences given to MOTIFS-IN, using the UPSTREAM-SEQUENCES-OF e-lambda.
- Associate the appropriate gene with the upstream sequence by using the LABELED option of UPSTREAM-SEQUENCES-OF.
- MOTIFS-IN is not very intelligent. You need to tell it that your sequences are DNA, not protein, by using its DNA option.
- Use the MULTIPLE-HITS-OK option. Unless you do this, MOTIFS-IN will find only one match per sequence. As you know, the cI region has multiple (3) repressor-binding sites.

SQ10. Would MOTIFS-IN be able to find the lambda operators shown above? Why (not)? Try it!

MOTIFS-IN returns the best alignments it found in the Results pane, but it also displays a complicated page produced by MEME describing these alignments. Let's go through that page. Scroll the display to Motif 1, what MOTIFS-IN considers to be the best motif it found. You'll see several sections:

Section 1: **Statistics** (labeled MOTIF *n*), containing width, sites, llr, and E-value
Section 2: **Simplified position-specific probability matrix**
Section 3: **Information content**
Section 4: **Consensus sequence**
Section 5: **Sites**
Section 6: **Block diagram**

Section 7: **Blocks**
Section 8: **Position-specific scoring matrix**
Section 9: **Position-specific probability matrix**

Consider Section 1 in light of what follows it.

SQ11. What do you think width and sites mean? Be sure to account for the specific numbers given.

SQ12. In plain English, what do you think E-value means? Again use the meaning of E-value in Blast for inspiration.

Consider Section 5 (**Sites**).

SQ13. What do you make of the column labeled Start? Test your hypothesis by displaying the SEQUENCE-OF the UPSTREAM-SEQUENCE-OF the first gene in the Sites list (under Name). Where do you find the sequence shown in the first line under Sites? How do the sequences under Sites relate to the sequences in the Blocks section? How do they relate to the sequences returned in the Results Pane?

SQ14. How would you construct the Multilvel Consensus Sequence section from the Sites section? What rules would you follow?

SQ15. Qualitatively, how do you relate the height of the bars in the Information content section to the Sites section? (You can get a quantitative rendering of the same information using the INFORMATION-OF function in BioBIKE, giving it the FIRST set of sequences in the result of MOTIFS-IN)

As for the rest of the sections,… wait until BNFO601 Integrated Bioinformatics.

How do you know if MOTIFS-IN did what you wanted it to? Since you're (wisely) using a case where you know what you're looking for, you need to identify the genes whose upstream sequences should contain the repressor-binding sites.

SQ16. What genes are those – what are their BioBIKE names? (If you don't recall, I advise a refresher of the previous notes on regulatory sequences).

SQ17. Do any of the three motifs returned by MOTIFS-IN incorporate upstream sequences from the two genes you identified in SQ16? Which one? How many sites were found in front of those two genes? Is it the number you expected?

Another check on your sanity is whether the motif you identified in SQ17 gives the sequence you expected (the repressor-binding sites – operators – shown in the table on p.1).

SQ18. Do you find the three operators in one of the motifs shown in the MEME page? If not, look carefully. Any way you could be fooled?

III. Search for regulatory sequences when they are not already known

MOTIFS-IN seems to be effective so far, so let's give it a sterner test. Get the following article:

> Gomathi NS, Sameer H, Kumar V, Balaji S, Azger Dustackeer VN, Narayanan PR (2007). In silico analysis of mycobacteriophage Che12 genome: Characterization of genes required to lysogenise *Mycobacterium tuberculosis*. Computational Biology and Chemistry 31:82-91

Most of this article is not of any concern to us at the moment, but near the end of the Results section, Gomathi et al bring up the subject of stoperators. I've never heard of this term. I believe

it's jargon used only by mycobacteriophage people. But never mind the term, the function is clear and very important. In order to affect gene expression, the repressor must bind to DNA near the operons it regulates. Since Che12 has only one repressor, it must bind to a single sequence (more or less). If it regulates several operons, there must be several copies of this sequence in the phage.

Conclusion, there must be many copies (or near copies) of a DNA sequence the size of a protein footprint (typically 6 to 15 nucleotides). Gomathi et al's Fig. 8 shows many copies of a 13-nt sequence.

SQ19. How many copies? Child's play! You know how to get a count of a specific sequence in a genome!

SQ20. How did Gomathi et al find these copies?

Well, yes. They cheated. They knew that Che12 is similar to another mycobacterial phage L5 and that L5 was known to have these sequences. Given the sequence, it was easy to search for it.

But suppose you didn't know the specific sequence? You know that SOME sequence is in multiple copies because you found a repressor protein, and it must repress something. But how do you find a repeated sequence if you don't know what the sequence is? Sounds like a job for MOTIFS-IN. Try it.

SQ21. In BioBIKE, DEFINE a set consisting of all the UPSTREAM-SEQUENCES-OF Che12, setting a MINIMUM-SIZE of 15 and LABELing the sequences with the names of the genes to which they're attached. Then use that set as the argument for MOTIFS-IN telling the function (which is rather stupid) that the sequences are DNA. After perhaps 10-20 seconds, you should receive back a window that contains three motifs (none of them guaranteed to be any good). Scroll down to Motif 1. What is its E-value? What do you suppose that E-value means? What is the sequence? How does it relate to the sequence Gomathi et al shows in their Fig. 8? (Don't give up on it too soon)

SQ22. Why isn't the number of matches you found in SQ21 the same as the number of matches you found in SQ19?