HW1: BLAST

CS 182/282 Spring 2024

Released: Tuesday, January 30th, 2024

Due: 11:59pm on Thursday, February 8th, 2024

Overview

BLAST (*B*asic *L*ocal *A*lignment *S*earch *T*ool) is one of the most widely-used bioinformatics algorithms. It compares a query sequence with target sequences stored in a database or library to identify homologous regions, which has earned it a reputation as the "Google of biological research". In this HW, you will run some simple BLAST queries on both nucleotide and protein sequences, reflect on a source of bias when working with sequence data, and learn about one of the earliest protein sequence alignments.

This assignment is worth a total of 50 points.

Handin

Submit your answers to the following problems as a PDF on Gradescope. You may include images of hand-drawn diagrams if necessary, but all written responses must be typed up. Do not include any identifying information on your handin.

P1: BLASTing Basic Nucleotide Sequences (6 points)

This problem will walk you through a basic BLAST query. Please follow the directions carefully to ensure that you arrive at the intended result.

- Visit NCBI BLAST.
- Click on **Nucleotide BLAST**.
- In the "Enter Query Sequence" panel, input the following DNA sequence:

```
AAACCTCCGATGCGTTATCAGTCGCACGTTTCCGCACCTGTCACGGTCGGGGCTTGGCGCTGCTGAGGGACACGCGTGAACCGA
GGAGACGGCAAGGACATCGCCGGAGATCCGCGCCTCGACAACGAGAAACCCTGCTAGACAGACCGCTCGAGAACACCGCAGCGA
GATTCAGCGTGCGGCAAAATGCGGCTTTTGACGAGAGTGCTGCTGGTGTCTCTTCTCACTCTGTCCTTGGTGGTGTCCGGACTG
GCCTGCGGTCCTGGCAGAGGCTACGGCAGAAGAAGACATCCGAAGAAGCTGACACCTCTCGCCTACAAGCAGTTCATACCTAAT
GTCGCGGAGAAGACCTTAGGGGCCAGCGGCAGATACGAGGGCAAGATAACGCGCAATTCGGAGAGATTTAAAGAACTTACTCCA
AATTACAATCCCGACATTATCTTTAAGGATGAGGAGAACACGGGAGCGGACAGGCTCATGACACAGAGATGCAAAGACAAGCTG
AACTCGCTGGCCATCTCTGTAATGAACCACTGGCCAGGGGTTAAGCTGCGTGTGACAGAGGGCTGGGATGAGGACGGTCACCAT
TTTGAAGAATCA
```

- In the "Choose Search Set" panel, select **Standard databases (nr etc.): Nucleotide collection (nr/nt)** under "Database" and leave all other fields blank.
- In the "Program Selection" panel, select **Highly similar sequences (megablast)**.
- Click **BLAST**!
- Scroll down past the graphic summary of results to the "Descriptions" section.

- Now, answer the following questions:
    1. What protein does this gene encode?
    2. What organism does it come from? Give the common name.

P2: Primer Design and Protein BLAST (12 points)

Primers are short strands of DNA which are used to bind to specific regions of the genome for a variety of biological applications. One such application is the *polymerase chain reaction*, or PCR, which amplifies target DNA through the use of forward and reverse primers which "sandwich" the region of interest. BLAST can be used to find whether PCR primers will produce off-target effects (i.e., bind erroneously to undesired regions of the genome).

The following sequences represent forward and reverse primers (both 5'-3') for a specific gene:

`AAGCAGTAGCTACCCGCGGGA` `CGGCTTCTTCCTGAGGACCT`

Here, we will use a tool dedicated to finding all primer hits: Primer BLAST. Paste each primer into its corresponding field, and simply hit "Get Primers"! (Leave all the default inputs unchanged.) The output will be a set of "templates" to which these primers align.

Once you have obtained your results, answer the following questions:

1. What gene will these primers amplify?
2. How long is the *amplicon* targeted by these primers? Do these primers amplify the entire gene?
3. Could this primer pair have any off-target effects (in humans specifically)?

Now that you've identified the gene of interest, we'll run BLAST on the corresponding protein sequence! We will use a particular *domain* (a highly conserved amino acid sequence that can be found in many other organisms with similar proteins) to search for similarity across many species. Follow the instructions below:

- Find the protein above on UniProt. Make sure to select the *Homo sapiens* version of the protein.
    4. What is the UniProt ID for this protein? (If it's not immediately obvious, check the URL.)
- Scroll down to the "Features" section (under the "Family and Domains" heading).
- Select the second domain listed.
    5. What is the name of this domain?
- You can run BLAST directly from UniProt! Select **BLOSUM-62** under "Matrix" and click **Run BLAST**.
    6. What two non-human organisms appear first? Give their common names.

P3: BLAST Runtime Analysis (14 points)

We now consider the time complexity of BLAST. Assuming that a query sequence has m characters and the database has n characters total, give a runtime analysis (using big-O notation) for each of the three main parts of the algorithm:

1. Generating seed words (k-mers) from the query and the alphabet Σ
2. Filtering seed words by checking whether each k-mer from our alphabet has an ungapped alignment score greater than some threshold, T, for some k-mer in the query.
3. Searching the database to identify seed "hits"
4. Extending all seed "hits" to identify maximal alignments

Note that these are all extreme worst-case scenarios and look bad on paper, but BLAST's real advantage is in employing heuristics to substantially undercut these maximum runtimes (i.e., leveraging the relatively small alphabet sizes of DNA/protein sequences).

Now answer the following questions:

5. What is the full runtime of BLAST (in terms of all variables defined above)? Note: *If we make the assumption that the size of the query is less than the size of the database ($m < n$), we don't have to include the complexity of step 2 from above.*
6. What is the runtime of a more naive search algorithm we saw in CS181 (i.e., ungapped local alignment between the query and the database)?
7. Assuming k and $|\Sigma|$ are constant in general, under what conditions will BLAST outperform this naive algorithm? Note: *No formal proof is required here, you can explain your answer in paragraph form.*

P4: Karlin-Altschul Statistics (12 Points)

Karlin-Altschul statistics are governed by the equation

$$E = Kmne^{-\lambda S}$$

where E is the expected number of random alignments with a given score S, K is a normalizing constant, m and n correspond to the lengths defined above, and λS is the normalized score of the given alignment. Based on the equation above, answer the following questions:

1. Suppose you have some sequence of length m you are trying to align with a subsection of a reference genome of length n. Describe how doubling the search space in this reference would affect E. Give an intuitive justification for your answer.
2. Describe how doubling the score would affect E. Give an intuitive justification for your answer.
3. Describe how decreasing the score by a constant value would affect E. Give an intuitive justification for your answer.

Recall also that the **bit score** is defined as

$$S_b = \frac{\lambda Y_{\text{MAX}} - \ln K}{\ln 2}$$

where Y_{MAX} is the maximum score attained during BLAST's alignment.

4. Why is it useful to express E in terms of bit score, as opposed to the original formulation?

5. Show that if $S = Y_{\text{MAX}}$, then the formulation of the Karlin-Altschul equation above is equivalent to:

$$E = mn \cdot 2^{-S_b}$$

Hint: Express λS in terms of the bit score, and then substitute into the original Karlin-Altschul equation.

P5: Reference Bias (6 points)

Recall from CSCI 1810 the problem of *reference bias* in the human genome: during the assembly of the human genome, most sequences were obtained from individuals of European descent. In 1810, you described some of the consequences of this bias, but here we will study them in greater detail. Visit this link to read about the latest and greatest assembly of the human reference genome, GRCh38.

1. Processing of genomic and epigenomic data typically proceeds by alignment of short $\sim 10^2$ bp reads to the full human genome (by a program similar to BLAST). With a single reference sequence derived from a mostly homogeneous population, many reads may fail to map to the correct region or to the genome at all. Describe a property of a gene/genomic region which may make it difficult to map correctly. (Consider the KIR gene cluster as an example.)
2. Discuss two potential issues that may arise from failed mappings as described in P4.1. (*Hint*: In what ways might a read fail to map? What are the consequences of each?)
3. Explain how the most recent assembly, GRCh38, attempts to combat reference bias. Describe at least one ongoing challenge with their solution.

If you're interested, read this paper for a complicated technical solution addressing reference bias, demonstrated on the major histocompatibility complex (MHC) region of the genome!