CS313 Exercise 4 Cover Page

Name(s): _____

In the *TIME* column, please estimate the time you spent on the parts of this exercise. Please try to be as accurate as possible; this information will help us to design future exercises.

PART	TIME	SCORE
Exercise		

Task 1: Identification of Similar Genes in Other Organisms

In this task, you will be performing a BLAST search with some yeast genes. Go to the BLAST website at NCBI <http://blast.ncbi.nlm.nih.gov/Blast.cgi>. Since you will be searching for amino acid sequences (rather than DNA sequences) that are similar to the yeast gene amino acid sequences, you should opt to perform a protein blast search.

Just upstream of the yeast hexokinase gene that you looked at a couple of weeks ago in Exercise #1 is a gene named *rpn12* that codes for a component of the 26S proteasome lid. Retrieve the protein sequence for *RPN12* from the yeast genome database and perform a BLAST search with this sequence.

Excluding the hits to *Saccharomyces* (which is the genus that we obtained the sequence from), what is the range in bit scores among the hits reported by BLAST?

Excluding the hits to *Saccharomyces* (which is the genus that we obtained the sequence from), what is the range in E-values among the hits reported by BLAST?

Excluding the hits to *Saccharomyces* (which is the genus that we obtained the sequence from), what is the range in percent identities among the hits reported by BLAST?

Excluding the hits to *Saccharomyces* (which is the genus that we obtained the sequence from), what is the range in percent positives among the hits reported by BLAST?

Excluding the hits to *Saccharomyces* (which is the genus that we obtained the sequence from), what is the range in percent gaps among the hits reported by BLAST?

Click on the Taxonomy tab near the top of the BLAST results. Based on the Lineage Report, to what kingdom of organisms do most of the BLAST hits for *RPN12* correspond?

Are there any BLAST hits to organisms outside this kingdom?

In the yeast genome, just downstream of the hexokinase gene, *hxk1*, is a genomic element named *YFR054C* that is annotated as a putative protein of unknown function. Retrieve the protein sequence for *YFR054C* from the yeast genome database and perform a BLAST search with this sequence, **but in the BLAST "Algorithm parameters" set the "Expect threshold" to 40 and the Word size to 3.**

Excluding the hits to *Saccharomyces* (which is the genus that we obtained the sequence from), what is the range in bit scores among the hits reported by BLAST?

Excluding the hits to *Saccharomyces* (which is the genus that we obtained the sequence from), what is the range in E-values among the hits reported by BLAST?

Excluding the hits to *Saccharomyces* (which is the genus that we obtained the sequence from), what is the range in percent identities among the hits reported by BLAST?

Excluding the hits to *Saccharomyces* (which is the genus that we obtained the sequence from), what is the range in percent positives among the hits reported by BLAST?

Excluding the hits to *Saccharomyces* (which is the genus that we obtained the sequence from), what is the range in percent gaps among the hits reported by BLAST?

Click on the Taxonomy tab near the top of the BLAST results. To what kingdom of organisms do most of the BLAST hits for *YFR054C* correspond?

Do you think that all of the hits reported by BLAST for *YFR054C* are homologs of *YFR054C*? Why or why not?

Task 2: Assessing BLAST Results

One application of BLAST searches is the identification of likely homologs for a query gene sequence. While we cannot say for sure that two sequences are homologous (without a time machine, it is difficult to be certain about ancestral sequence relationships), BLAST is generally effective at finding similar sequences to a query sequence, and based on the similarity of sequences we may hypothesize that the sequences are homologous if the sequences are significantly more similar than we would expect by chance. **However, it should be noted that BLAST is a *heuristic* search process, which means that BLAST may not report all similar (or homologous) sequences to a query sequence.**

Let the *sensitivity* of a BLAST search be defined as the percentage of actual homologs of a query sequence reported by the BLAST search. For example, suppose we have a database of 200 target sequences. Let's assume that some query gene has 20 homologous sequences in the database, and a BLAST search using the query gene sequence results in a list of 50 significantly similar sequences to the query sequence. Among the 50 significantly similar sequences reported by BLAST are 15 of the 20 homologs in addition to 35 other non-homologous sequences. Since the BLAST search reports 15 of the 20 actual homologs, the sensitivity of this search would be 75%.

Let the *specificity* of a BLAST search be defined as the percentage of non-homologous sequences to a query sequence that are *not* reported by the BLAST search. For example, suppose we have a database of 200 target sequences. Let's assume that some query gene has 20 homologous sequences in the database, and a BLAST search using the query gene sequence results in a list of 50 significantly similar sequences to the query sequence. Among the 50 significantly similar sequences reported by BLAST are 15 of the 20 homologs in addition to 35 other non-homologous sequences. Since there are 180 non-homologous sequences in the database and since the BLAST search does not report 145 of the 180 non-homologous sequences (the search does report 35 of the 180 non-homologous sequences), the specificity of this search would be 145/180 = 80.6%.

A useful property of heuristic algorithms is having high sensitivity (i.e., reporting many of the true homologous relationships) and having high specificity (i.e., not reporting many false-positive, non-homologous relationships). However, often there is a trade-off between sensitivity and specificity - an algorithm's sensitivity can be increased at the cost of lower specificity, or an algorithm's specificity can be increased at the cost of lower sensitivity.

As an extreme example, imagine that for a query sequence with 20 homologs, a BLAST search suggests six million significantly similar sequences to the query sequence. Assuming the 20 true homologous sequences are among the list of six million sequences reported by BLAST, the sensitivity of the search is 20/20 = 100%, which is excellent. However, the specificity of the search is very poor because many spurious, non-homologous sequences were reported. Such a search would not be useful because it would not help distinguish the 20 actual homologs from the roughly six million non-homologous sequences.

Alternatively, imagine that for a query sequence with 20 homologs, a BLAST search suggests zero significantly similar sequences to the query sequence. The sensitivity of the BLAST search would be poor, 0/20 = 0%. However, the specificity of the search would be excellent, 100%, because no non-homologous sequences were reported. Such a search would not be useful because it would not provide insights into the 20 actual homologs of the query sequence.

The search parameters of a heuristic algorithm often allow users to explore the trade-off between sensitivity and specificity. For a given query sequence, parameters that cause more results to be reported generally lead to higher sensitivity at the cost of lower specificity. Parameters that cause fewer results to be reported generally lead to higher specificity at the cost of lower sensitivity.

Suppose we have the following amino acid query sequence, "RKYVHFQNS". Give an example of an amino acid target sequence that is more than 60% identical to the query sequence, but that a BLAST search would not identify with its default parameter settings. How could you change the parameter settings so that BLAST indeed would identify your target sequence?

If you increase the "Max target sequences" parameter value, how will the sensitivity and specificity of a BLAST search be affected?

If you decrease the "Word size" parameter value, how will the sensitivity and specificity of a BLAST search be affected?

If you increase the "Expect threshold" (i.e., the E-value), how will the sensitivity and specificity of a BLAST search be affected?

If you change the scoring "Matrix" from BLOSUM62 to PAM30, how will the sensitivity and specificity of a BLAST search be affected?

Task 3: Alignment of Random Sequences to Estimate Significance

Suppose that 10,000 pairs of random DNA sequences are generated. Each DNA sequence is 50 nucleotides in length and has a GC content of 50%. The optimal local pairwise alignment is then computed for the 10,000 pairs of sequences, so that 10,000 optimal local alignment scores are obtained. The following URL illustrates this data:

http://cs.wellesley.edu/~cs313/exercises/Exercise4/Exercise4_Distribution.html

On this webpage, the second column of the table indicates how many of the 10,000 alignments had the score indicated in the first column of the table. The third column of the table indicates the percentage of the 10,000 alignments with the score indicated in the first column of the table. The final column of the table indicates the estimated percentage of the 10,000 alignments with the score indicated in the first column of the table. The estimation in the fourth column is obtained by computing the mean and standard deviation of the 10,000 alignment scores and using this mean and standard deviation to define an extreme value distribution that approximates the actual 10,000 scores. The graph at the top of the page is a plot of the data in the table - the bar graph reflects the third column in the table and the line graph reflects the fourth column in the data. As an example, in the graph, the largest value in the bar graph occurs at a score of 26. In the table, it can be seen that 863 of the 10,000 alignments (or 8.63%) achieved a score of 26.

Suppose someone gave you a pair of DNA sequences, each of length 50 nucleotides and GC content of 50%, and you then determined the optimal local alignment score for the pair. Based on the abovementioned webpage, what is the minimum optimal local alignment score the pair of sequences could have such that the similarity of the pair of sequences would have a significance of $p <= 0.05$.

Suppose someone gave you a pair of DNA sequences, each of length 50 nucleotides and GC content of 50%, and you then determined the optimal local alignment score for the pair. Based on the abovementioned webpage, what is the minimum optimal local alignment score the pair of sequences could have such that the similarity of the pair of sequences would have a significance of $p <= 0.01$.

Suppose someone gave you a pair of DNA sequences, each of length 50 nucleotides and GC content of 50%, and you then determined an optimal local alignment score of 50 for the pair. Using the abovementioned webpage, calculate the percentage of the 10,000 random alignments with optimal local alignment score greater than or equal to 50. What is the *p*-value for a score of 50, i.e., what is the likelihood that two random sequences would have a score of at least 50?

Rather than determine the *percentage of random alignments* with optimal local alignment score at least 50, the *p*-value for a score of 50 can be estimated from an extreme value distribution that approximates the data. The *p*-value can be calculated using formula (1) below, which corresponds to an extreme value distribution,

$$1-e^{-e^{-\frac{x-u}{\beta}}} \qquad (1)$$

where μ is a location parameter and β is a scale parameter. The location and scale parameters can be computed from the mean and standard deviation of the abovementioned 10,000 alignment scores as follows,

$$\mu = mean - (0.5772 * \beta) \qquad \beta = standardDeviation * \sqrt{6} / \pi$$

Assuming the abovementioned 10,000 alignment scores have a mean of 31 and a standard deviation of 7, based on an extreme value distribution approximating the 10,000 alignment scores, what is the *p*-value for an optimal local alignment score of x=50?

Suppose we have previously determined the location parameter, μ, and the scale parameter, β, of an extreme value distribution approximating the optimal alignment scores of random sequences 50 nucleotides in length with GC content of 50%. What is an advantage of estimating the *p*-value of new alignments using formula (1) above rather than by generating and aligning thousands of pairs of random sequences?

Task 4: E-values

In Task 3 above, pairs of sequences were aligned. For a given pair of sequences, let us call the first sequence the *query* sequence and let us call the second sequence the *target* sequence. In Task 3, each query sequence was aligned with only one target sequence. Task 3 is analogous to comparing a query sequence to a database consisting of a *single* target sequence. If we have a database containing not one target sequence but millions of target sequences, when comparing a query sequence to the database we may align the query sequence to *many* target sequences in the database.

When comparing a query sequence to a database of target sequences, the *p*-value of an alignment score, *S*, is the likelihood that the query sequence, when aligned to the target sequences in a comparable random database, would produce one or more query:target alignment scores greater than or equal to *S*. In other words, when aligning a single query sequence to:

- a *single* target sequence, the *p*-value of the alignment score, *S*, is the likelihood that the two sequences have an alignment score of at least *S* by chance.
- *multiple* target sequences, the *p*-value of the alignment score, *S*, for a given query:target alignment is the likelihood that at least one of the multiple query:target alignments would produce an alignment score of at least *S* by chance.

The *E*-value is related to the *p*-value by the following formula:

$$E = -\ln(1-p)$$

If the alignment score of a query sequence to one of the target sequences in a database is *S*, the *E*-value of the alignment score is the expected number of alignments with score at least *S* when comparing the query sequence to a random database. Thus, an *E*-value of 1.0 for an alignment score, *S*, indicates that we can expect 1 query:target alignment to have a score of at least *S* by chance. An *E*-value of 2.0 for an alignment score, *S*, indicates that, when comparing a query sequence with a database, we can expect 2 alignments to have a score of at least *S* by chance. An *E*-value of 5.0 for an alignment score, *S*, indicates that we can expect 5 alignments to the database with score of at least *S* by chance.

Suppose a query sequence is aligned to a database of target sequences, and the alignment score of the query sequence to one of the target sequences is 55, which has a significance of $p = 0.95$. When aligning the query sequence to all of the target sequences, how many out of all the query:target alignment scores can we expect to have a score of at least 55?

Suppose a query sequence is aligned to a database of target sequences, and the alignment score of the query sequence to one of the target sequences is 60, which has an *E*-value of 0.5. What is the *p*-value of this alignment score?

Below is a table showing the relationship between *E*-values and *p*-values over a range of scores. While reading the table, you may notice that at low values of *E* and *p*, the scores are quite similar, while they diverge at higher values.

Relationship of *E* to *p*-values in BLAST

E	*p*
10	0.99995460
5	0.99326205
2	0.86466472
1	0.63212056
0.1	0.09516258
0.05	0.04877058
0.001	0.00099950
0.0001	0.00010000

When performing BLAST searches, an important step is assessing the significance of your results. In experimental biology, a *p*-value of 0.05 or lower is often considered statistically significant (note that this corresponds to a 5% or less chance that random data would achieve such a result or better). However, for BLAST searches, researchers sometimes use more stringent criteria for accepting results as non-random. For example, genome databases often use *E*-value cut-offs in the 10^{-3} to 10^{-6} range for annotations. Why might researchers use such stringent criteria for BLAST analyses?

Suppose a query sequence is aligned to a database of target sequences, and the alignment score of the query sequence to one of the target sequences is 45, which has an *E*-value of 6.0. Now suppose that, over time, new sequences are added to the database until the number of target sequences in the database has *doubled*. If the query sequence is aligned to the bigger (doubled) database, and the alignment score of the query sequence to one of the target sequences is 45, what would be the *E*-value for this alignment score of 45?

Suppose a query sequence is aligned to a database of target sequences, and the alignment score of the query sequence to one of the target sequences is 45, which has a *p*-value of p_1. Now suppose that, over time, new sequences are added to the database until the number of target sequences in the database has *doubled*. When the query sequence is aligned to the bigger (doubled) database, the alignment score of the query sequence to

one of the target sequences is 45, which has a p-value of p_2. What is the relationship, if any, between p_1 and p_2? In other words, can we tell which is bigger or are they the same?