EE 372: Data Science for High-Throughput Sequencing

Assignment 3

Due: June 1 in class

Question I: Minhashing

In class we discussed briefly how minhashing can be used for aligning reads with large error rates. In this question, we will explore the minhashing concept.

1. We can describe a read as a set of unique overlapping k-mers, and we would expect similar reads to have similar sets. Write a function that takes k and a read as inputs and outputs a dictionary indicating which of the 4^k k-mers are in the read.

2. If we think of each read as a set of k-mers, a natural metric of similarity between two reads R_1 and R_2 is the Jaccard similarity, which is defined as

$$J(R_1, R_2) = \frac{|R_1 \cap R_2|}{|R_1| + |R_2| - |R_1 \cap R_2|}$$

   Explain how this metric captures similarity between two sets and how you might use this metric to align reads (2-3 sentences). Compute the minimum and maximum possible Jaccard similarity between any two sets.

3. Write a function to compute the Jaccard similarity between two dictionaries outputted by your function from part 1. Using this function and the one you wrote for part 1, compute the Jaccard similarity between the reads `CATGGACCGACCAG` and `GCAGTACCGATCGT` for $k = 3$. What is the runtime complexity of your function? If you have N reads of length L each, what is the worst-case runtime complexity for computing the Jaccard similarity between every pair of reads?

4. Suppose you have a function that hashes a k-mer to a value between 1 and 4^k. For minhashing, you would use this hash function to map each unique k-mer in a read to an index, ultimately returning the smallest index. Prove that the probability that two sets will generate the same minhash index is equal to their Jaccard similarity.

5. In practice, we would use multiple H hash functions to compute multiple minhash indices for R_1 and R_2. Write down an estimator that uses the H minhash indices to estimate $J(R_1, R_2)$. What is the runtime complexity of obtaining this estimation? How does this compare to the runtime you obtained for part 2?

Question II: Haplotype phasing coverage

In this problem we examine a simplified version of the haplotype assembly problem. We assume that a genome has n SNPs, and each SNP is heterozygous. Every mate-pair read covers a pair of adjacent SNPs.

1. We assume that we know the exact position of the read on the genome by aligning the read to the genome. How many different types of reads do we obtain (restricting ourselves to their values at the SNP positions) if there were no errors?
2. Given that we obtain N mate-pair reads, argue that for large N and n, the number of reads covering each adjacent pair of points $\sim$ Poisson(N/n).
3. If each position of each mate-pair read has an error rate of $\delta \in [0,1]$, argue that the number of erroneous pairwise measurements $\sim$ Poisson$(2\delta(1-\delta)N/n)$.
4. Given that there are m reads covering a pair of consecutive positions, argue that the probability that a majority of the reads being wrong is approximately $\exp\left(-mD\left(\frac{1}{2}||2\delta(1-\delta)\right)\right)$ where $D(\frac{1}{2}||2\delta(1-\delta))$ represents the KL divergence between a Bernoulli$(\frac{1}{2})$ distribution and a Bernoulli$(2\delta(1-\delta))$ distribution. *Hint*: The probability of Binomial$(m,q) > \frac{1}{2} = \exp(-mD(\frac{1}{2}||q))$.
5. Using earlier parts or otherwise, compute an upper bound on the average probability of error in estimating the parity between SNR i and $i+1$, in terms of N, n and δ.
6. Our final goal is to phase the **entire** genome correctly. Using part 6 or otherwise, compute a bound on the probability of error in phasing the entire genome.
7. For a given desired probability of phasing error ϵ, use the bound in part 7 to give an expression for N^*, the number of mate-pair reads needed.
8. For $\epsilon = \delta = 0.01$, plot N^* as a function of n, the number of SNP's. How does N^* scale with n as n grows? Linearly, sublinearly or superlinearly? Can you give an intuitive explanation for your answer?
9. For $\epsilon = 0.1$, $n = 100,000$, plot N^* as a function of δ, the read error rate.

Question III: RNA-seq Quantification

In class we discussed the basic EM algorithm for RNA-seq quantification in the simple case when the transcript lengths are all the same and there are no errors in the read. In this question, we will consider extensions to unequal transcript lengths and read errors. We start with the same RNA-seq model as discussed in class.

1. Instead of equal transcript length ℓ, let us now consider the case when the transcript lengths are $\ell_1, \ell_2, \dots \ell_K$. The reads are still error-free.
   (a) Develop the log likelihood model.
   (b) Derive the EM iterative algorithm for this model, specializing from the general EM algorithm discussed in the lecture.
2. Now suppose the reads have errors: each base is read correctly with probability $1 - \delta$ and incorrectly with probability δ, and if incorrect the base can be read equally likely as any of the three other possibilities.
   (a) Generalize the log likelihood model in Part 1 to this case.
   (b) Derive the EM iterative algorithm for this model, again specializing from the general EM algorithm.
   (c) Suppose the alignment tool at your disposal can compute all exact alignments and all approximate alignments up to one base different. If the error rate δ is small such that the chance of a read having two or more errors is negligible, explain how you would use your alignment tool in implementing the EM algorithm derived in the previous part.