

QBIO 478 - Extra Credit Assignment

Name: Kiley Huffman

Student ID: 5582-0328-36

Problem 1 (1 point)

Some k-mer matching methods allow for wildcard patterns where multiple different letters may match at the same position. For example the wildcard pattern A{CG}T will match ACT and AGT. For the pattern G{AC}T{TG}AT, calculate the probability of a 5-letter word from a genome matching, and the expected number of matches for a genome generated by the random nucleotide generator model and is 1000 letters long where (a) each nucleotide has equal probability of being generated (e.g. $p_A = 0.25, p_C = 0.25, p_G = 0.25, p_T = 0.25$), and (b) where $p_A = 0.1, p_C = 0.2, p_G = 0.3, p_T = 0.4$. State any approximations you use. Show your work clearly, and briefly explain why the expected number of matches changes between the two genomes.

Pattern: G{AC}T{TG}AT, Possible k-mers with the given pattern = $1 * 2 * 1 * 2 * 1 * 1 = 4$

Number of possible 6-letter windows in the genome = $1000 - 6 + 1 = 995$

Assumptions:

We assume that each nucleotide in the genome is generated independently of the others.

We assume that the overlapping windows (e.g., positions 1–6, 2–7, etc.) in the 1000-letter genome are independent.

We assume that only exact matches to the 4 valid 6-mers are considered.

(a) Each nucleotide has equal probability of being generated:

Since each base has an equal probability of occurring, the probability of one of the 4 possible k-mers being generated is:

$$p(\text{choosing one of the possible } k - \text{mers}) = 0.25^6 = 0.00024414$$

Since the probability of choosing the k-mer is the same for each of the 4 possible k-mers, the probability of generating one of any of the 4 valid k-mers:

$$p(\text{match}) = 4(0.25^6) = 0.00097656$$

And the expected number of matches for the given pattern is: $995 \cdot 0.00097656 = 0.9727$

(b) Where $p_A = 0.1$, $p_C = 0.2$, $p_G = 0.3$, $p_T = 0.4$:

The probability of generating a 6-letter word from a genome matching the pattern is:

$$p(\text{match}) = p_G * (p_A + p_C) * p_T * (p_T + p_G) * p_A * p_T$$

$$p(\text{match}) = (0.3)(0.1 + 0.2)(0.4)(0.4 + 0.3)(0.1)(0.4) = 0.001008$$

And the expected number of matches is: $995 \cdot 0.001008 = 1.003$

(c) Briefly explain why the expected number of matches changes between the two genomes:

In the uniform genome (part a) the nucleotides are all equally likely to occur ($p = 0.25$). In the biased genome (part b), nucleotides like T ($p = 0.4$) and G ($p = 0.3$) are more likely to occur than the others. Since the pattern $G\{AC\}T\{TG\}AT$ includes several Ts andGs, the overall probability of matching increases in the biased genome (part b), even though the number of matching patterns remains the same (there are still only 4 k-mers that match the wildcard pattern).

Problem 2 (1 point)

A sequencing instrument produces reads with an error probability of p . (a) Describe how sequencing errors can fragment assembly using de Bruijn graphs. (b) What is the probability of sequencing a k-mer with no errors? (c) Consider a 100-base read that has a sequencing error at the 50th position. If the read is used in an assembly using de Bruijn graphs with $k = 21$, how many k-mers will be affected by the error? The result of the error is to change the content of k-mers that overlap the error in the read from what was in the genome to a different k-mer. That new k-mer potentially is a k-mer that does not exist in the genome, or it is a k-mer that matches a different position in the genome from where the read was sampled. In humans, this is likely to happen in reads that overlap Alu sequences, with the explanation related to why all humans are 99.9% similar with respect to single-nucleotide variation. Provide a computational explanation for why this is the case.

(a) Describe how sequencing errors can fragment assembly using de Bruijn graphs:

In de Bruijn graphs, reads are broken into overlapping k-mers that form nodes and edges based on shared $(k-1)$ -mers. A single sequencing error introduces k-mers that differ from the correct sequence. These error k-mers do not match any other correct k-mer in the graph, resulting in dead-end paths (tips) or erroneous branches (bubbles) in the graph. These artifacts fragment the graph since correct paths are interrupted by incorrect low-frequency k-mers that do not connect to true neighboring k-mers. In short, sequencing errors introduce unique or incorrect k-mers,

which create artifacts (tips/bubbles) in the graph and cause fragmentation by breaking otherwise continuous assembly paths.

(b) What is the probability of sequencing a k-mer with no errors?:

Let p be the error probability per base, and assume errors are independent.

Then, the probability that one base is correct $= 1 - p$

Thus, for a k-mer of length k , the probability that all bases are correct is:

$$p(k - \text{mer has no errors}) = (1 - p)^k$$

(c) Consider a 100-base read that has a sequencing error at the 50th position. If the read is used in an assembly using de Bruijn graphs with $k = 21$, how many k-mers will be affected by the error?:

A k-mer is affected by an error if it overlaps the error position. Since each k-mer spans k positions, a k-mer will include position 50 if it starts at positions $50 - k + 1$ to 50. This means k-mers starting at positions 30 through 50 will be affected. From position 30 to 50, there are 21 possible k-mers, thus 21 k-mers will be affected by the error at position 50.

(d) Provide a computational explanation for why this is the case:

Alu elements are short repetitive sequences (~300 bp) that appear numerous times in the human genome. Thus, a single error in a read overlapping an Alu may alter k-mers so that they match a different copy of the repeat. This causes ambiguous placement of reads during assembly. Since many Alu sequences are highly similar, even a single base error may cause a read to appear to come from a different location. This increases the risk of misassembly or fragmentation in repetitive regions. In other words, sequencing errors that overlap Alu regions generate k-mers that could falsely match many similar genomic regions. This misguides the de Bruijn graph traversal and leads to incorrect or ambiguous assembly.

Problem 3 (1 point)

The telomere-to-telomere sequencing of the human genome was accomplished during the COVID-19 lockdown. Describe the algorithmic and technical challenges that needed to happen in order for this genome to be sequenced, or some of the biological discoveries that were made possible using the T2T genome. Your answer should be no more than four paragraphs. Your answer should reference two figures from the paper, and include a screenshot of the part of the figure that your answer addresses.

One major challenge the T2T team faced was that repetitive DNA (such as satellite DNA, centromeres, telomeres, and segmental duplications) makes up over 50% of the human genome. These sequences are nearly identical over long stretches, making it extremely difficult to determine their correct order and copy number using traditional short-read sequencing technologies. To solve this, the T2T team leveraged ultra-long Oxford Nanopore reads (hundreds of kilobases in length) to span entire repetitive regions, and combined these with high-accuracy PacBio HiFi reads (~15–20 kb) to polish the sequence.

Additionally, new assemblers such as Shasta, Flye, and HiCanu were developed specifically for long-read data. These tools included error correction, repeat resolution strategies, and graph-based assembly models tailored to long, noisy inputs. These technological innovations were critical for assembling and validating all 23 human chromosomes, including the previously unresolved Y chromosome and centromeric regions, which are shown to be fully resolved in **Figure 3** of the T2T paper. This figure illustrates the complete chromosomal assemblies with coverage extending across previously unresolvable areas.

Another key challenge was ensuring the structural correctness of the genome assembly. To address this, the T2T team used multiple orthogonal data types, including Bionano optical maps, Hi-C chromatin conformation capture, and Strand-seq, to scaffold, validate, and detect any potential mis-assemblies. For correcting base-level errors in the final consensus, tools like PEPPER-Margin-DeepVariant were employed. Additional algorithmic advances included identifying and cataloging higher-order repeat (HOR) units, applying graph tiling methods to model complex sequences, and conducting manual curation based on coverage patterns and repeat structure validation.

As a result of these innovations, the T2T project produced the first-ever complete assembly of human centromeres. The assembly revealed over 200 million base pairs of novel sequences, including previously undetected genes, gene duplications, and regulatory elements. These sequences were previously inaccessible and omitted from the standard reference genome. For example, **Figure 4** in the T2T paper showcases the complete assembly of centromeric alpha satellite arrays for each chromosome. This figure demonstrates how the T2T genome clarified the structure and sequence variability of these repeats across the genome, offering new insights into the repetitive architecture of human DNA.

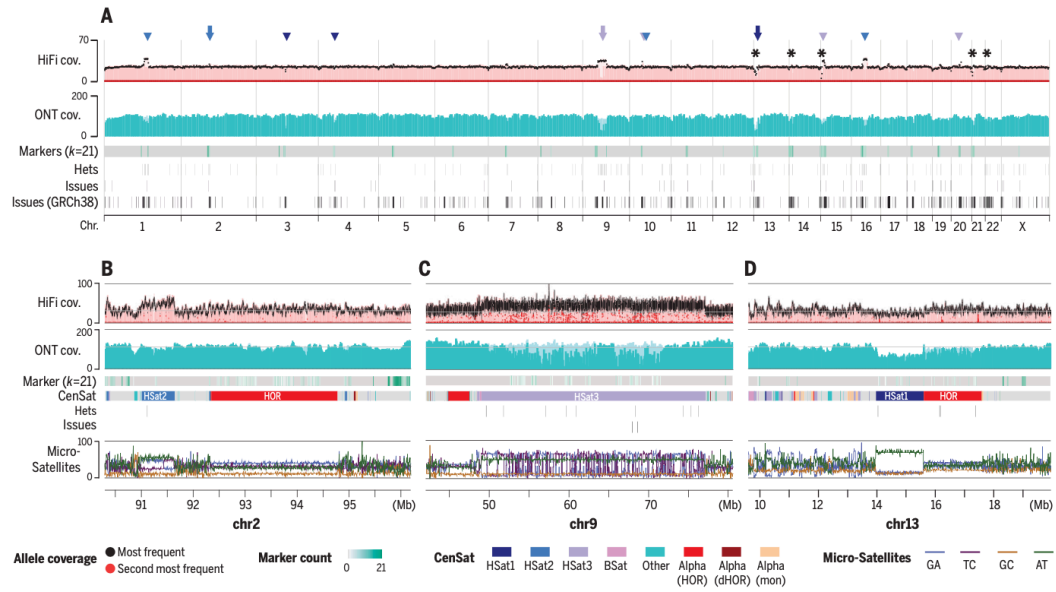


Fig. 3. Sequencing coverage and assembly validation.

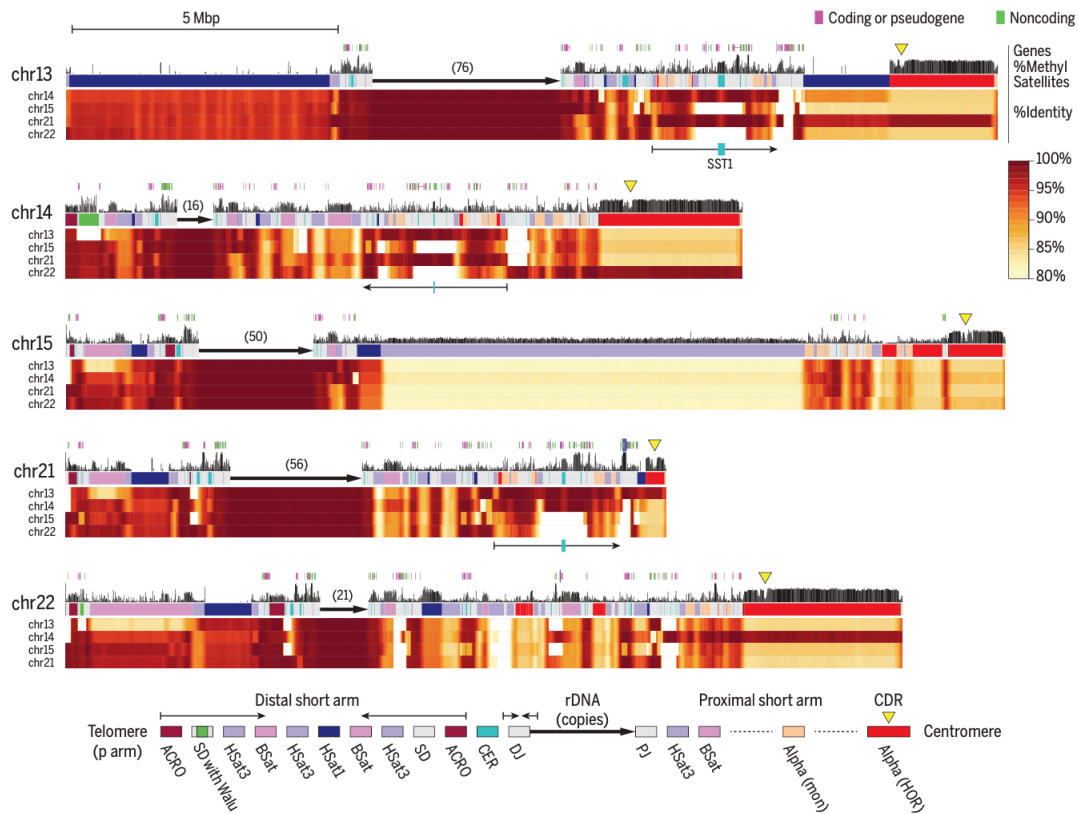


Fig. 4. Short arms of the acrocentric chromosomes.