Discussion 4 - pseudoalignment

Albert Xue

January 2023

Bolded terms are defined in the glossary.

1 Introduction

Alignment is the process of determining the origin of a read in the genome. When we align a read to a reference genome, we are attempting to determine from what exact location this read came from, given this reference genome.

But in RNA-seq, our concern is not the original position of the read within the genome. We are trying to quantify gene expression by counting mRNA transcripts; as a result, our primary concern is from what *gene* (***isoform***), rather than from what *position*, the read came from. **Pseudoalignment** is the analogous process. When we **pseudoalign** a read to a reference **transcriptome**, we are determining the gene (isoform) that generated this transcript.

There are complications and details that we will cover. But this is essentially what *kallisto* does: it takes reads from RNA-seq protocols, and determines what genes (isoforms) generated these transcripts.

Here are the key concepts of kallisto.

1. kallisto builds a **de Bruijn graph** from the reference transcriptome.

2. Because we want gene counts, we don't care where the read comes from exactly; we just care from which gene the read comes from. Actually, gene counts are too simplistic; genes have multiple **isoforms**. We actually care from which **isoform** the read comes from.

3. Paths in the de Bruijn graph map a sequence to an isoform.

4. Sometimes we can't determine exactly which isoform a read comes from. It's sufficient to determine the **equivalence class** for each read.

5. We only need to examine the "breakpoints" of the de Bruijn graph, where isoforms diverge from each other.

2 kallisto builds a de Bruijn graph from the reference transcriptome

A de Bruijn graph[1] is a type of graph representing a sequence or a string. Each node is a unique k**-mer**; an arrow between nodes represents a transition between adjacent k-mers. Maybe more

[1] When I took this class, literally all we did was build de Bruijn graphs.

intuitively, an arrow from node X to node Y represents sliding a window of size k one step across the original sequence.[2] Note that a de Bruijn graph is always specific to k, the length of the k-mer.[3]

Let's take a look at Figure 1. **Ignore the green path for now, and focus only on the path with blue and pink**. That path is the de Bruijn graph of the sequence ACATGTCCAGT with k-mer length 3.

Let's decompose this step by step. We have the sequence ACATGTCCAGT. How do we create the de Bruijn graph with $k = 3$?

First we create the set of 3-mers in ACATGTCCAGT. Fortunately, today they are all unique, and we obtain the set {ACA, CAT, ATG, TGT, GTC, TCC, CCA, CAG, AGT}. This set becomes our set of nodes.

Next, we draw an arrow between adjacent k-mers in the sequence. For example, we draw an arrow between node ACA and node CAT, then between CAT and ATG, etc.

That's the whole de Bruijn graph! Conceptually, a de Bruijn graph is a representation of text, or a string, into graph form. Our chosen sequence, ACATGTCCAGT, was an easy case because there are no repeated k-mers.

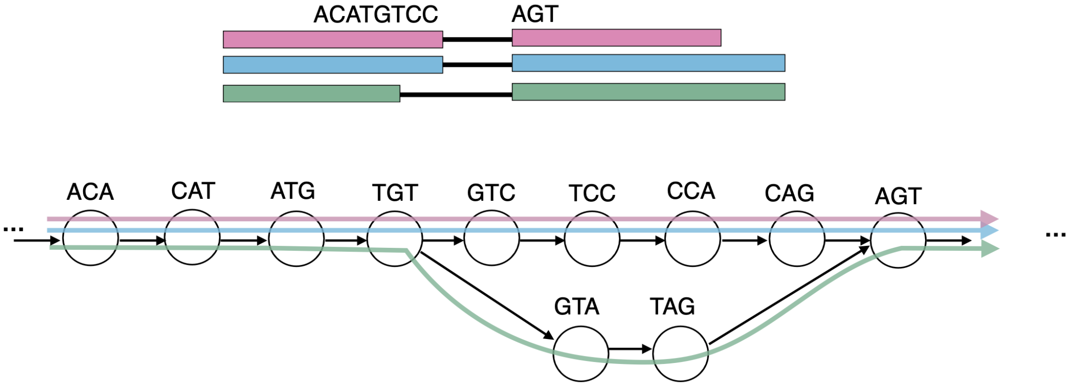

Figure 1: Chromatic paths are added onto the de Bruijn graph to represent different isoforms.

Question 1. *What is the de Bruijn graph of the sequence AAABBBBA with $k = 2$?*

Traditionally, when we build a de Bruijn graph we only retain information about the sequence. But Kallisto is interested in the gene/isoform that these transcripts come from, and so we need more information on top of our traditional de Bruijn graph.

Fortunately, we can do this in a relatively simple manner. From the reference, we know the sequences of all the isoforms (that's how we built the de Bruijn graph in the first place). We therefore know which k-mers map to which isoforms, and can build a **chromatic de Bruijn graph** that also includes isoform information. The easiest way to understand this is visually, as in Figure 1; each color represents an isoform, and is overlaid over the sequence in the De Bruijn graph.

Question 2. *What must be the sequence of the green isoform, given Figure 1?*

[2] If you're curious about more, see https://www.cs.jhu.edu/ langmea/resources/lecture_notes/assembly_dbg.pdf
[3] Why?

3 Equivalence classes on isoforms

Sometimes it's impossible to determine exactly which isoform a read came from. For example, in Figure 1, the given sequence ACATGTCCAGT could have feasibly come from either the pink or the green isoforms. In these cases, the best you can do is determine the **equivalence class** of isoforms from which the read could have come from. An **equivalence class** is defined in the following manner.

Definition 1. *Given a set S and equivalence relation $\sim$, the equivalence class of an element $a \in S$, denoted by $[a]$, is the set*

$$\{x \in S \mid x \sim a\}. \tag{1}$$

An **equivalence relation** is defined in the following manner:

Definition 2. *An equivalence relation is a binary relation that is reflexive, symmetric and transitive.*

What does all this mean? An equivalence relation $\sim$ is formally defining the vague intuition of "equivalent to". An equivalence class, then, is a set in which all elements in the set are "equivalent to" each other. Saying that $\sim$ is reflexive means that we can write $x \sim x$, and say that x is equivalent to itself. $\sim$ symmetric means that if $x \sim y$, then $y \sim x$; in words, if x is equivalent to y, then y is equivalent to x. $\sim$ transitive means that if $x \sim y$ and $y \sim z$, then $x \sim z$. Together, these three properties are sufficient to define a closed set E such that $\forall x, y \in E$, $x \sim y$.

What are our equivalence classes in Figure 1? Intuitively, they are the two sets {pink, blue} and {green}.

The process of kallisto is therefore as follows. For each read, we follow the read's transcript in the de Bruijn graph for all k-mers in the transcript. At each k-mer, we can identify the corresponding node in the de Bruijn graph. At each node/k-mer in the de Bruijn graph, we can identify the isoforms compatible with that node/k-mer. The isoforms compatible with all nodes/k-mers in the read will then be our return set of equivalence classes on reads.

4 The chromatic de Bruijn graph can be compressed into breakpoints

Most of Figure 1 is redundant. The nodes ACA, CAT, and ATG give us the same equivalence class, which is {pink, blue, green}, so checking their sequence information at all three nodes is on some level a waste of time.[4]

The solution is to "skip ahead" to the nodes that really matter. You can think of these nodes as the breakpoints in the graph, or the nodes that redefine equivalence classes. In Figure 1, that would be the two nodes GTC and GTA. If we observe "GTC" at that k-mer, our read is within the {pink, blue} equivalence class. Conversely, if we observe "GTA" at that k-mer, we know that our read is within the equivalence class {green}.

Question 3. *What will our compressed de Bruijn graph look like from Figure 1?*

Question 4. *What if there's an error in the read?*

Question 5. *What if there's an error in the reference transcriptome?*

Question 6. *Using pseudoalignment, map the read GCGCTAG to the isoforms*

[4] Also, you're more likely to see an error! And that screws everything up. More on that later.

1. $ATAAGCGCGCTAGCT$

2. $ATGCGCTAGCT$

3. $ATAAGCGCGCTA$

4. $ATGCGCT$

5 Glossary

1. **pseudoalignment** the process of determining the gene or isoforms that a read originated from, instead of the specific locations in the genome.
2. **transcriptome** the set of all possible mRNA transcripts that can be produced by an organism
3. **de Bruijn graph** a graph that encodes a string. Each node is a unique k-mer, and each arrow represents adjacency between k-mers in the string.
4. **isoform** Different forms of the mRNA transcript of a single gene can arise through differential inclusion of exons and introns, or alternative splicing.
5. **equivalence class** Given a set S and equivalence relation $\sim$, the equivalence class of an element $a \in S$, denoted by $[a]$, is the set
$$\{x \in S \mid x \sim a\}. \tag{2}$$
6. ***k*-mer** a subset of k adjacent letters in a string.
7. **chromatic de Bruijn graph** a de Bruijn graph whose nodes also contain information about compatible isoforms.