COM SCI C121 Week 5

Aidan Jan

May 2, 2024

De Brujin Graphs Review

To make from read:

1. Sample every 3-mer from the read.
2. Sample Left and Right 2-mers from each 3-mer.
3. On the graph, each 2-mer is a node, and 3-mers are the links between the nodes of the left and right 2-mers

We cannot go back from the De Brujin to the aligned genome. However, it is important to note that some Eulerian path (path that uses every link exactly once) would produce the original read.

Eulerian Walk Definitions and Statements

- Node is *balanced* if indegree equals outdegree
- Node is *semi-balanced* if indegree differs from outdegree by 1
- Graph is *connected* if each node can be reached by some other node
- *Eulerian walk* visits each edge exactly once
- Not all graphs hae Eulerian walks. Graphs that do are *Eulerian*.
- A directed, connected graph is Eulerian if and only if it has at most 2 semi-balanced nodes and all other nodes are balanced.

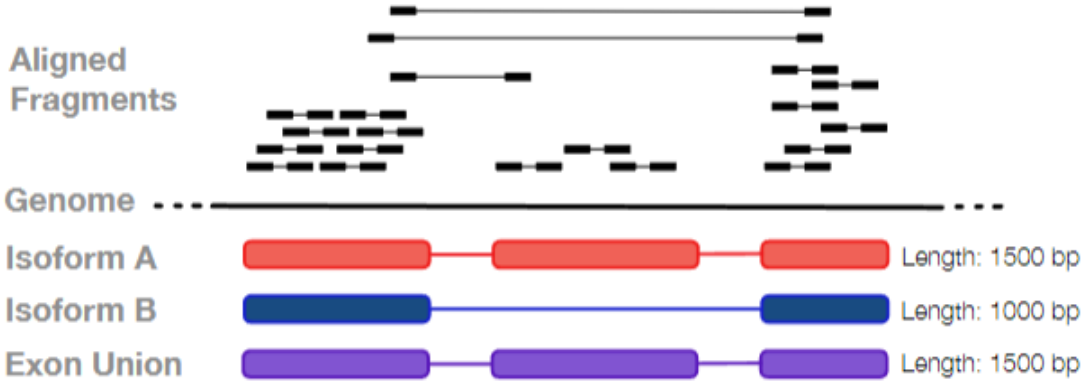
Attempt 2: Build the T-DBG

Consider the example:

```
         ACATACAT---ACA
RED      ########---###
GREEN    ########---###
BLUE     #####------###
```

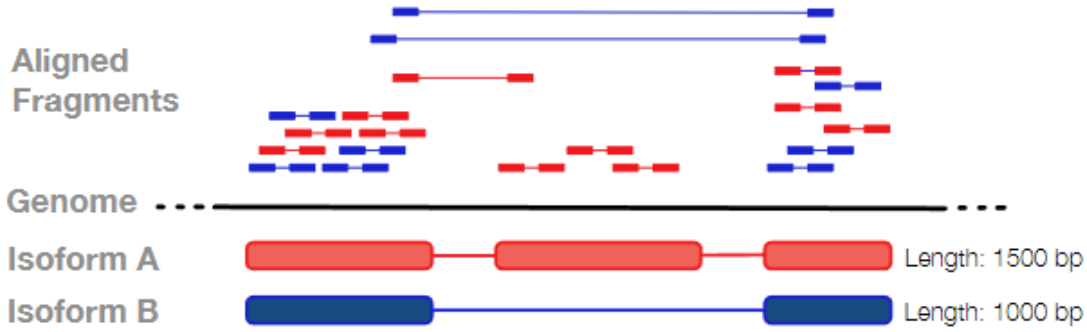
Where # denotes the strand having the base, and − denoting the base is absent on that strand.
For the example last week, we built a graph that was straightforward.
However, notice that some nodes were repeated this time.

From Alignments to Counting



Algorithm: Aggregate isoforms into a "gene" then count all fragments that overlap. There are several ways to aggregate the isoforms (see also Union-intersection)

Gene Counting May Be Misleading



Suppose that all of the fragments read have been aligned.

- Gene counting may be misleading because the fragments may have varying lengths, and so do the isoforms.

$$TPM_{true} = TPM_A + TPM_B \propto \frac{f_A}{l_A} + \frac{f_B}{l_B} = \frac{10}{1500} + \frac{10}{1000} = \boxed{\frac{1}{60}}$$

$$TPM_{union} \propto \frac{f_A + f_B}{l_A + l_B} = \frac{10}{1500} + \frac{20}{1500} = \boxed{\frac{1}{75}}$$

Note that $TPM_{union} \leq TPM$

Issues with Raw Counts

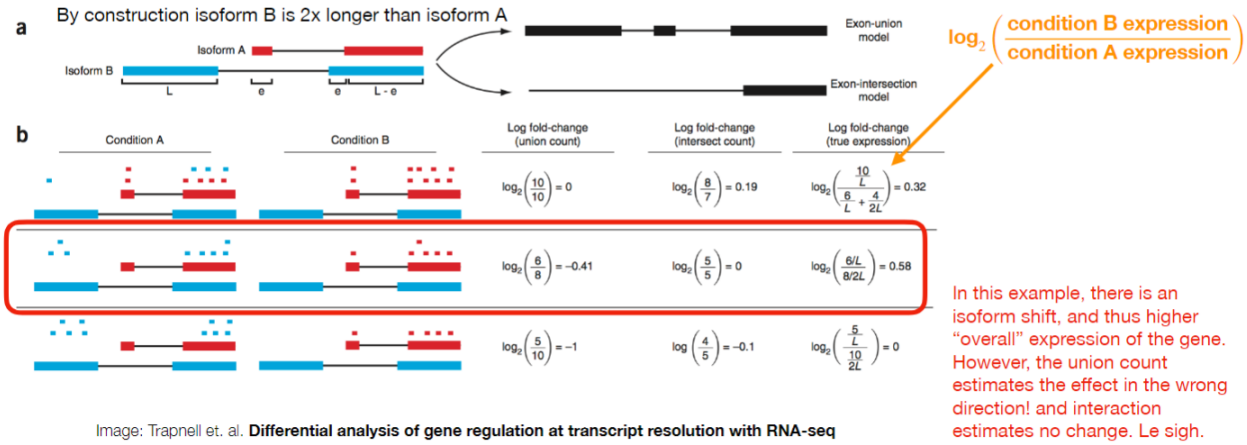
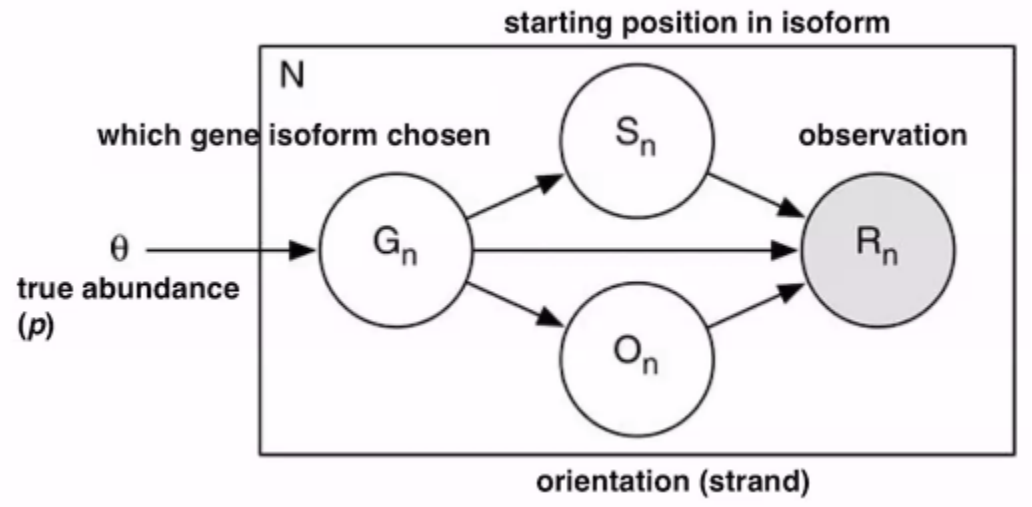


Image: Trapnell et. al. **Differential analysis of gene regulation at transcript resolution with RNA-seq**

- Because Isoform B is longer than Isoform A in reality, we cannot assume they are the same length.
- Condition A and Condition B are two separate experiments. For example, Condition A may be a control group and Condition B may be the genes with some sort of drug included.

RNA-seq, the full generative model



- θ represents the true abundance: that's what we are trying to solve for.
- In the generative case (going forwards through that model):
  - We first choose a gene.
  - We choose a position in the isoform to start
  - We choose orientation (forward or backward strands)
  - Those three factors gives us the observation. (some read that came from all the information)

Example: Suppose we generate some reads. We have three strands, a blue, red, and orange strand.

- G_1: red, $O_1 = +1$, $s_1 = \tilde{L}_{red}$ (red strand, forward, and the location starting is at the end of the strand)
- G_2: orange, $O_2 = +1$, $s_2 = \frac{\tilde{L}_{orange}}{3}$ (orange strand, forward, and location is 1/3 from the start of the strand)

Aside: RNA-seq, the Likelihood

- Our observations are the reads and everything else is a nuisance parameter, except theta (which we care about)
- What we observe is then:

$$P(r|\theta) = \prod_{n=1}^{N} \sum_{i=0}^{M} \theta_i P(r_n|G_n = i)$$

  - where N is the number of reads and M is the number of transcripts
- This equation is a pain in the ass to evaluate! Can we simplify this?

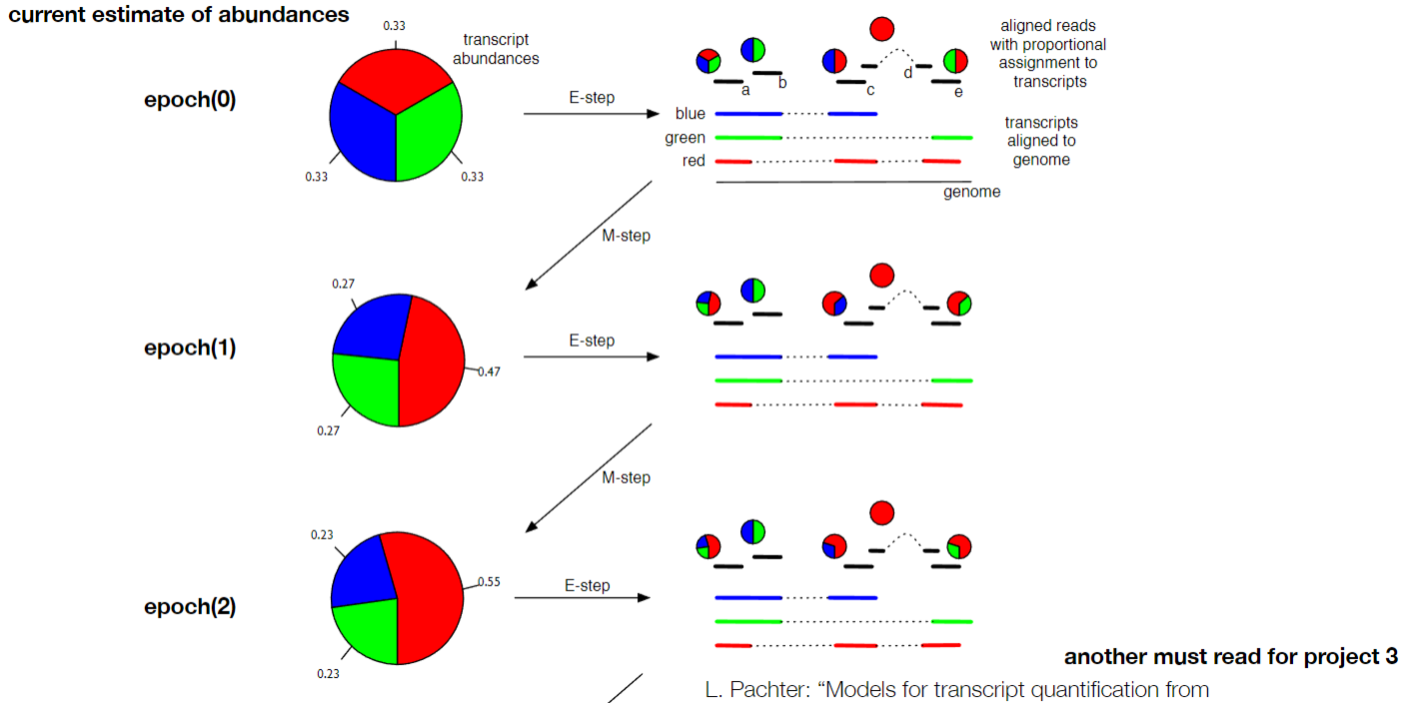
We can simplify the total likelihood by **read mapping**.

$$P(r|\theta) = \prod_{n=1}^{N} \sum_{i=0}^{M} \theta_i P(r_n|G_n = i) \rightarrow P(r|\theta) \approx \prod_{n=1}^{N} \sum_{(i,j)\in\pi_n^x} \frac{\theta_i}{l_i} P(r_n|Z_{nij} = 1)$$

- $z_{nij} = 1$ if $(i, j) = (s_n, o_n)$
- This approximation allows us to only sum over "compatible" alignments.

Isoform Abundance Estimation

- We have an unobserved variable (expression) that we wish to estimate
  - Set up a model and estimate it using the expectation-maximization (EM) algorithm
- Step 1: (Expectation) Given some abundances, estimate the probability of each read mapping to each transcript
  - We assume we know the actual data. (Pretend we know something based on the data.)
- Step 2: (Maximization) Update the abundances by redistributing the reads
  - We use our assumptions to maximize some other function.
- Step 3: (Repeat) Go to Step 1 until convergence.
  - We use the information we gathered from maximizing the other function to make another guess about the data.
  - We iterate this until it converges.



- On Epoch 0, we assume that the transcript abundances are equal.
- On each E-step, we generate pie charts for each aligned read based on the probabilities.
- On each M-step, we update the abundances based on the aligned reads.
- The converged value is not necessarily perfectly accurate, but it gets good enough.

Notes:

- Notice that the reads on each step do not change, just the probability they are associated with.

EM Algorithm Math Setup

$$P(r|\theta) = \prod_{n=1}^{N} \sum_{i=0}^{M} P(O_n = i) P(r_n | O_n = i)$$

$$P(r, s, g) = \prod_{n=1}^{N} \sum_{(i,j) \in \pi} P(O_n = i) P(S_n = j | O_n = i) P(r_n | O_n = i, S_n = j)$$

- (i, j) in the summation is the alignment compatibility
- $z_{nij} = 1$ if $(i, j) = (O_n, S_n)$

Under the assumption of uniform start positions (e.g., each transcript has equal abundance)

$$P(S_n = j | O_n = i) = \frac{1}{l_i} \qquad P(O_n = i) = \theta_i$$

Therefore,

$$p(r, s, y, |z, \theta) = \prod_{n=1}^{N} \sum_{(i,j)\in\pi} \frac{O_i}{l_i} P(r_n|z_{nij} = 1)$$
$$= \prod_n \prod_i \prod_j \left(\frac{\theta_i}{l_i} P(r_n|z_{nij} = 1) \right)^{\mathbb{1}\{z_{nij}=1\}}$$

where

$$\mathbb{1}\{X = 1\} = \begin{cases} P(X = 1) & (i, j) \in \pi \\ 1 - P(X = 1) & \text{otherwise} \end{cases}$$

or in words, it represents the characteristic function of a set.