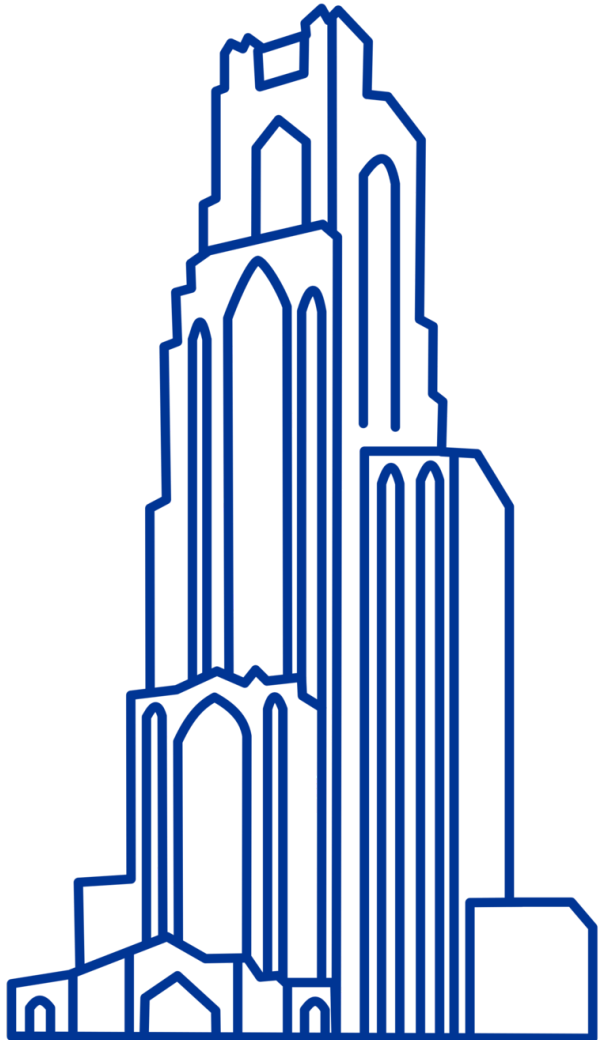


Computational Biology
(BIOSC 1540)

Lecture 09:
Quantification

Sep 24, 2024

University of Pittsburgh

Announcements

- A04 is due **Friday** by 11:59 pm
- Exam is next Thursday (Oct 3rd)

BIOSC 1540 - Computational Biology

Bioinformatics Exam

Oct 3, 2024

100 points

Please read the following instructions carefully before beginning your assessment.

- **Time limit:** You have 75 minutes to complete and turn in this assessment.
- **Open note:** You may use notes, but with the following restrictions:
  - Notes must be hand-written on either (1) paper or (2) a tablet with a stylus, then printed.
  - You may use a maximum of one sheet of 8.5 × 11 in. paper for notes (front and back allowed).
  - Notes must be your own work. Sharing or copying notes from others is strictly prohibited.
  - Your name must be clearly written on your notes.
- **No digital devices:** The use of digital devices, including calculators, is not allowed.
- **Submission requirements:** You must submit both your completed assessment and all notes used.

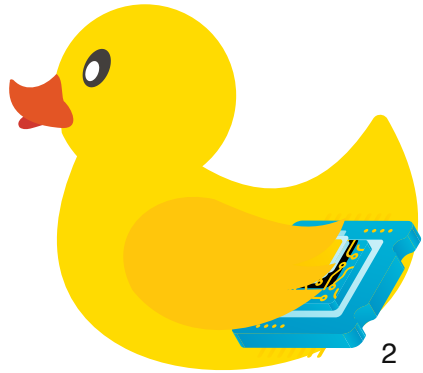
I agree to follow the above instructions. I affirm that all work on this assessment will be my own and that I will not give or receive any unauthorized assistance. To have your assessment graded, you must write your name, sign, and provide your student ID below.

_____ _____

Name Signature

_____

Student ID



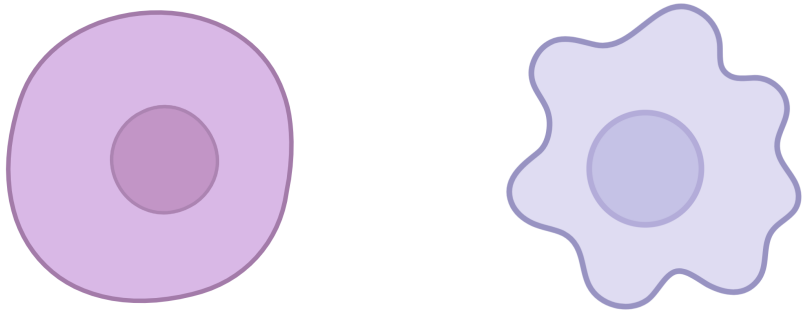
After today, you should be able to

1. **Discuss the importance of normalization and quantification in RNA-seq data analysis.**
2. Explain the relevance of pseudoalignment instead of read mapping.
3. Understand the purpose of Salmon's generative model.
4. Describe how salmon handles experimental biases in transcriptomics data.
5. Communicate the principles of inference in Salmon.

Let's pause and look at the big picture

Suppose we have isolated a
normal and cancerous cell

We want to identify possible drug targets
based on **overexpressed genes**



Normal Cancerous

We will use **transcriptomics**!

Defining our transcriptome

Let's simplify our problem to only **three transcripts**

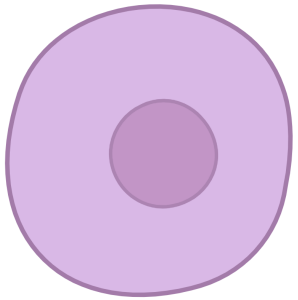
These represent the only mRNA transcripts we will find in our cells (i.e., the **transcriptome**)

t_1 t_2 t_3

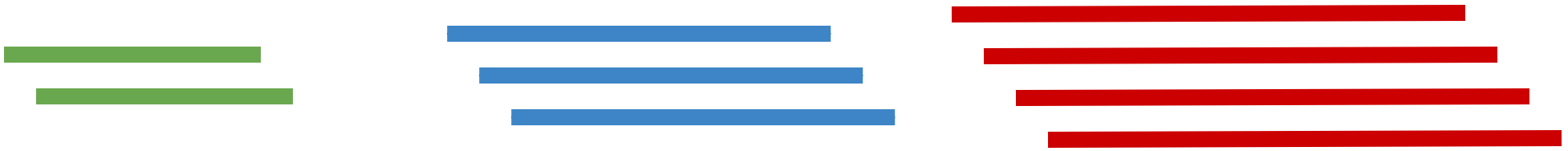
They have short, medium, and long **lengths**

$$l_3 > l_2 > l_1$$

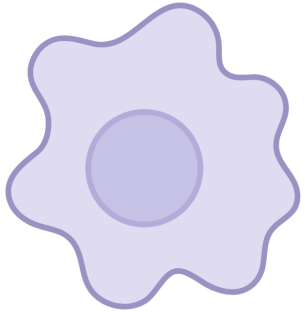
Defining our gene expression



Normal



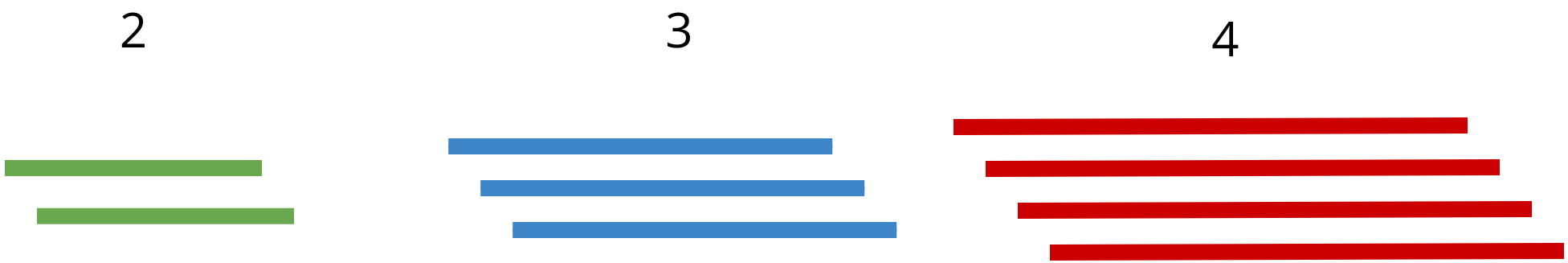
We have the following transcript distribution



Cancerous

More transcripts?	✓	✓	✓
Relative to others?	✗	✗	✓

We have to normalize our transcriptome before making comparisons



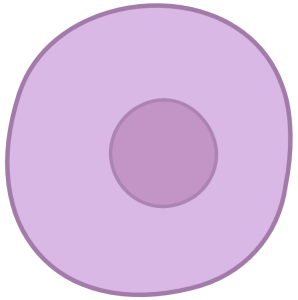
We can use transcripts fraction

$$\frac{2}{9} \approx 0.22$$

$$\frac{3}{9} \approx 0.33$$

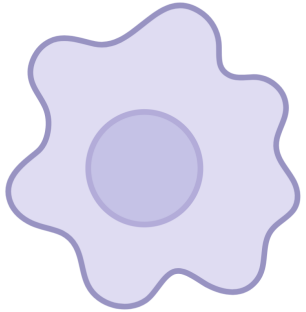
$$\frac{4}{9} \approx 0.44$$

Ratios are sensitive to total amount



Normal

$$\frac{2}{9} \approx 0.22 \qquad \frac{3}{9} \approx 0.33 \qquad \frac{4}{9} \approx 0.44$$



Cancerous

$$\frac{4}{15} \approx 0.27 \qquad \frac{6}{15} \approx 0.4 \qquad \frac{5}{15} \approx 0.33$$

Because the cancer cell is transcribing more overall, we still get changes across the board

Scaling data to "parts per million"

Real data has more than three transcripts and ratios are substantially smaller (e.g., 0.000001)

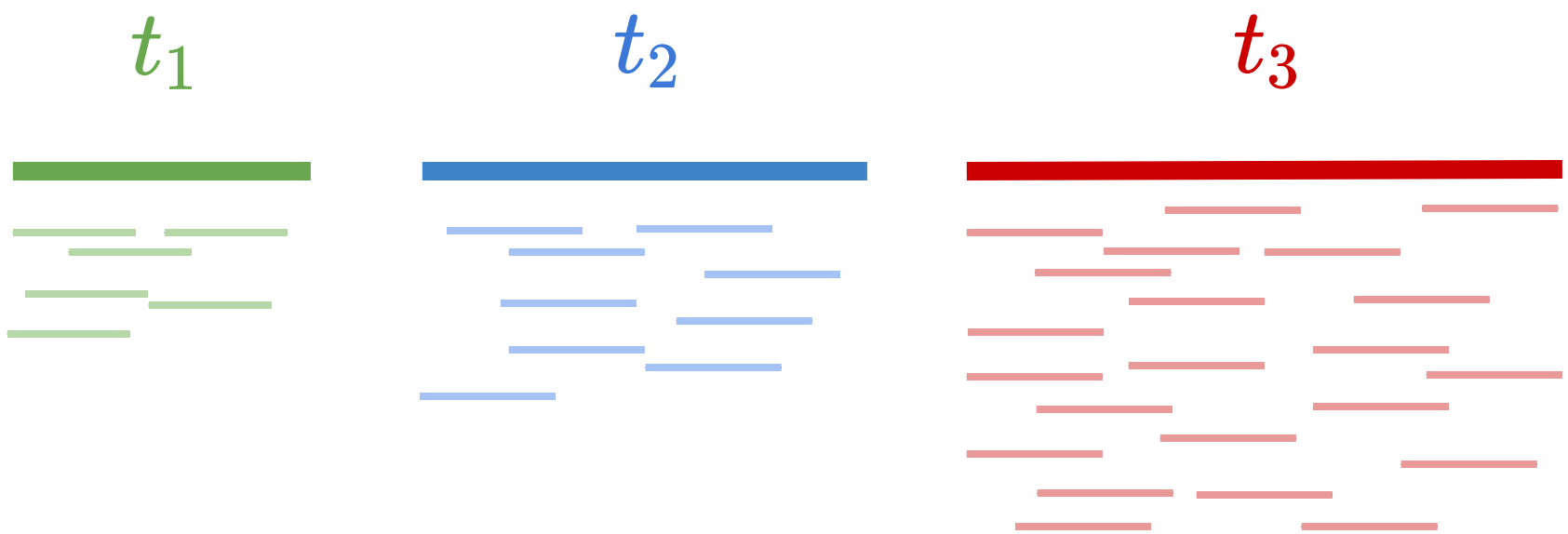
Small floats require high precision (i.e., float64) and thus memory

This can make computations and communications challenging, so we often scale everything to a million to use unsigned integers

$$\frac{t_i}{\sum t_i} \cdot 10^6$$

Transcript	Normal	Cancerous
1	222,222	266,666
2	333,333	400,000
3	444,444	333,333
Total	1,000,000	1,000,000

Wait, what about sequencing depth?



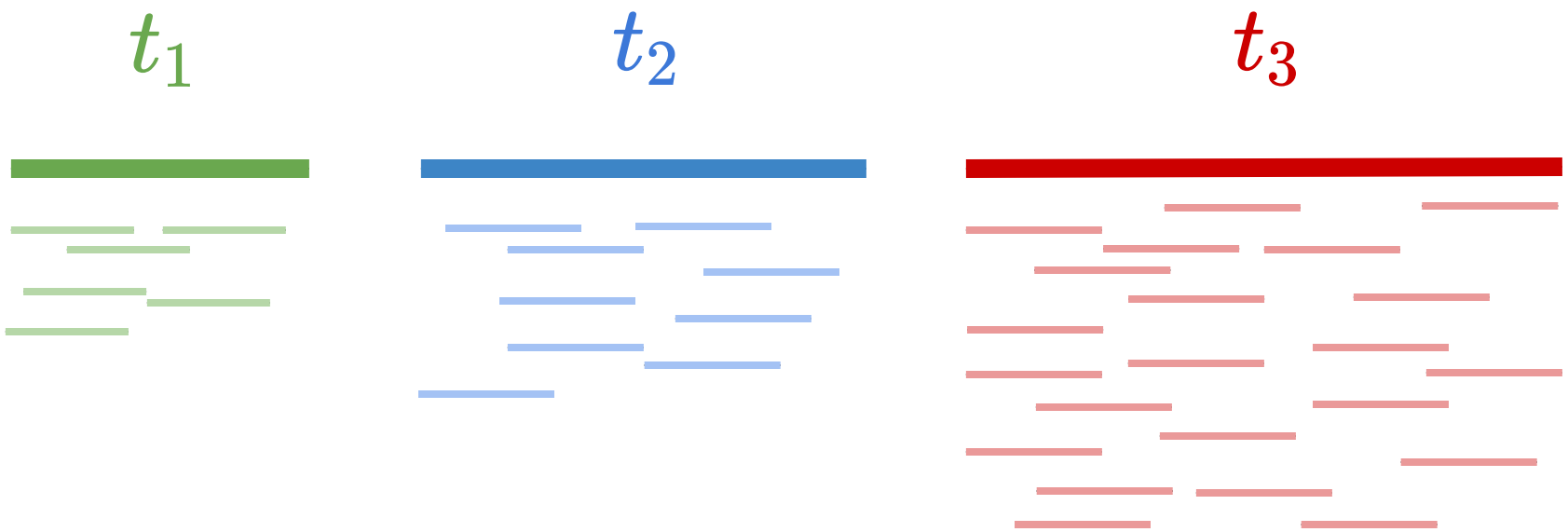
Longer transcripts will have more reads

Read per kilobase (RPK) corrects this experimental bias through normalization by gene length

$$RPK = \frac{\text{Read counts for gene}}{\text{Gene length in kilobases}}$$

(Length is usually just the exons)

RPK example



$$RPK_1 = \frac{\text{Read counts for gene}}{\text{Gene length in kilobases}}$$

Reads per kilobase of transcript per million reads mapped

$$\mathrm{RPKM} = 10^9 \frac{\text{Reads mapped to transcript}}{\text{Total reads} \cdot \text{Transcript length}}$$

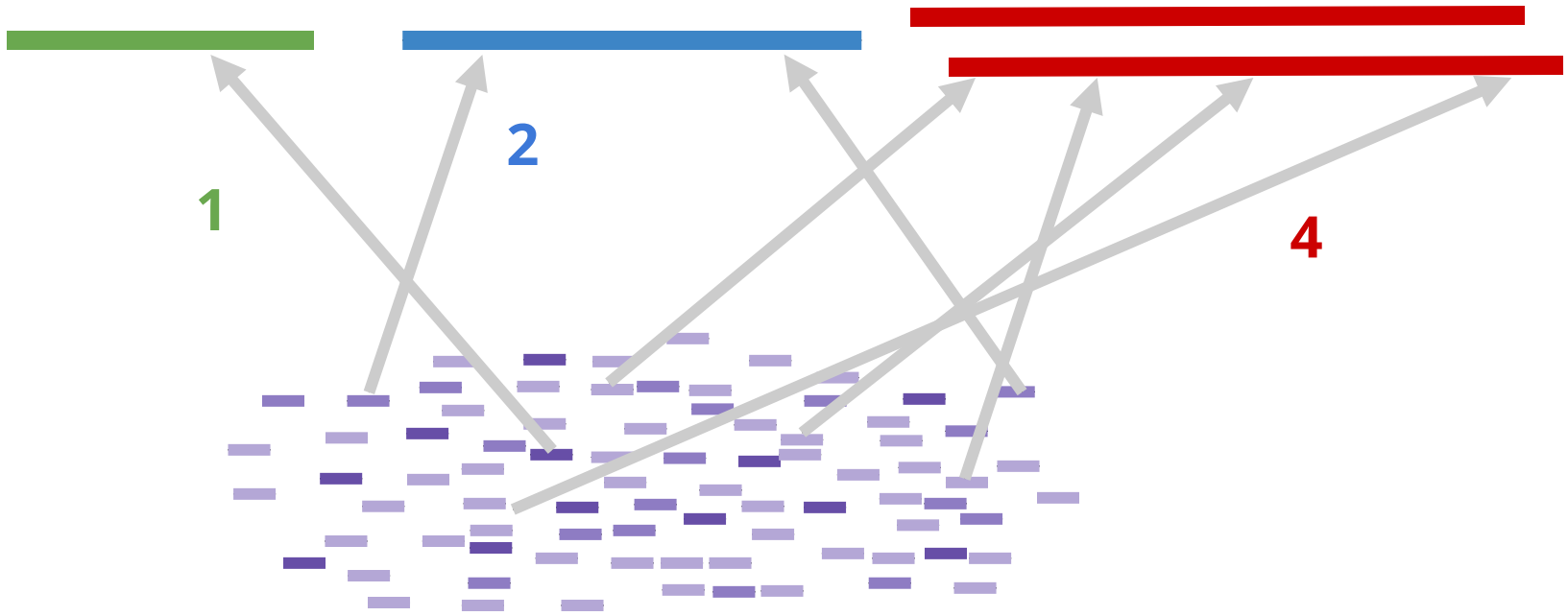
Transcripts per million

$$\mathrm{TPM} = 10^6 \frac{\mathrm{RPKM}}{\sum_i \mathrm{RPKM}_i}$$

After today, you should be able to

1. Discuss the importance of normalization and quantification in RNA-seq data analysis.
2. **Explain the relevance of pseudoalignment instead of read mapping.**
3. Understand the purpose of Salmon's generative model.
4. Describe how salmon handles experimental biases in transcriptomics data.
5. Communicate the principles of inference in Salmon.

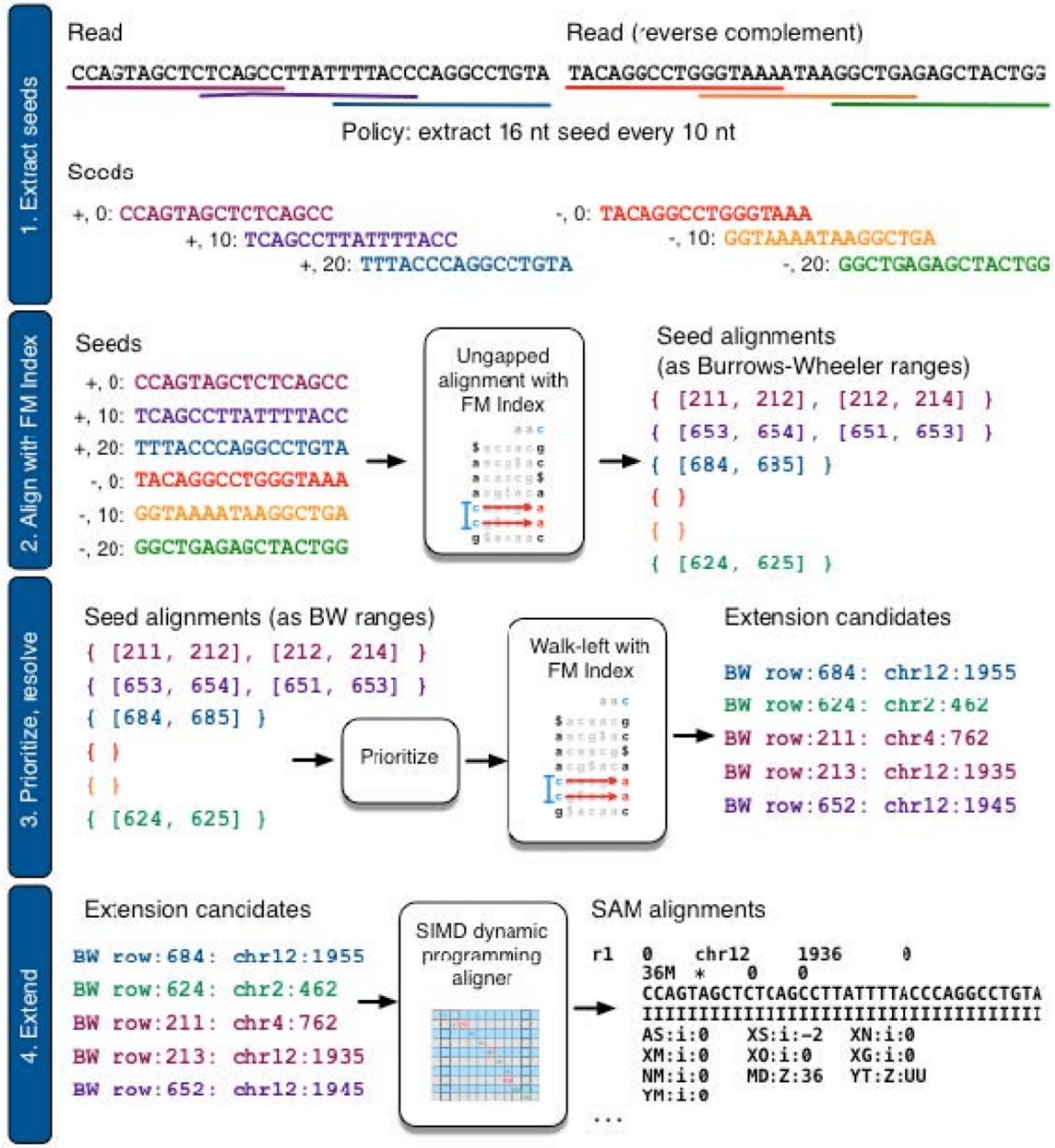
Traditional quantification uses read mapping



We assign each read to single transcript using our read mapping algorithms

Once aligned, we can count the number of mapped reads to each transcript

Bowtie 2 uses Burrows-Wheeler Transform to map and quantify reads

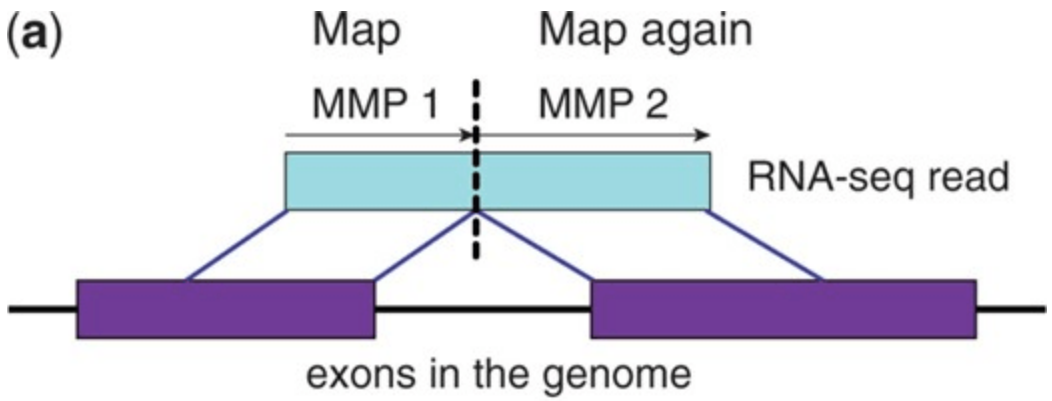


Spliced Transcripts Alignment to a Reference (STAR)

Maximum Mappable Prefix (MMP) approach for fast, accurate spliced alignments

Finds prefix that perfectly matches reference then repeats for unmatched regions

This automatically detects junctions instead of relying on databases



Alignment-based methods are computationally expensive

Suppose someone took library books (**transcripts**) and then shredded them (**reads**)

Alignment-based methods need to determine the **read's exact position in the transcript**

In the context of our analogy, we not only need **to find the book but which page it was from**

This takes a **long time**

Pseudoalignment finds which transcript, but not where

Identifies **which transcripts** are compatible with the read, skipping the precise location step

It does not worry about **where within that transcript it originated**

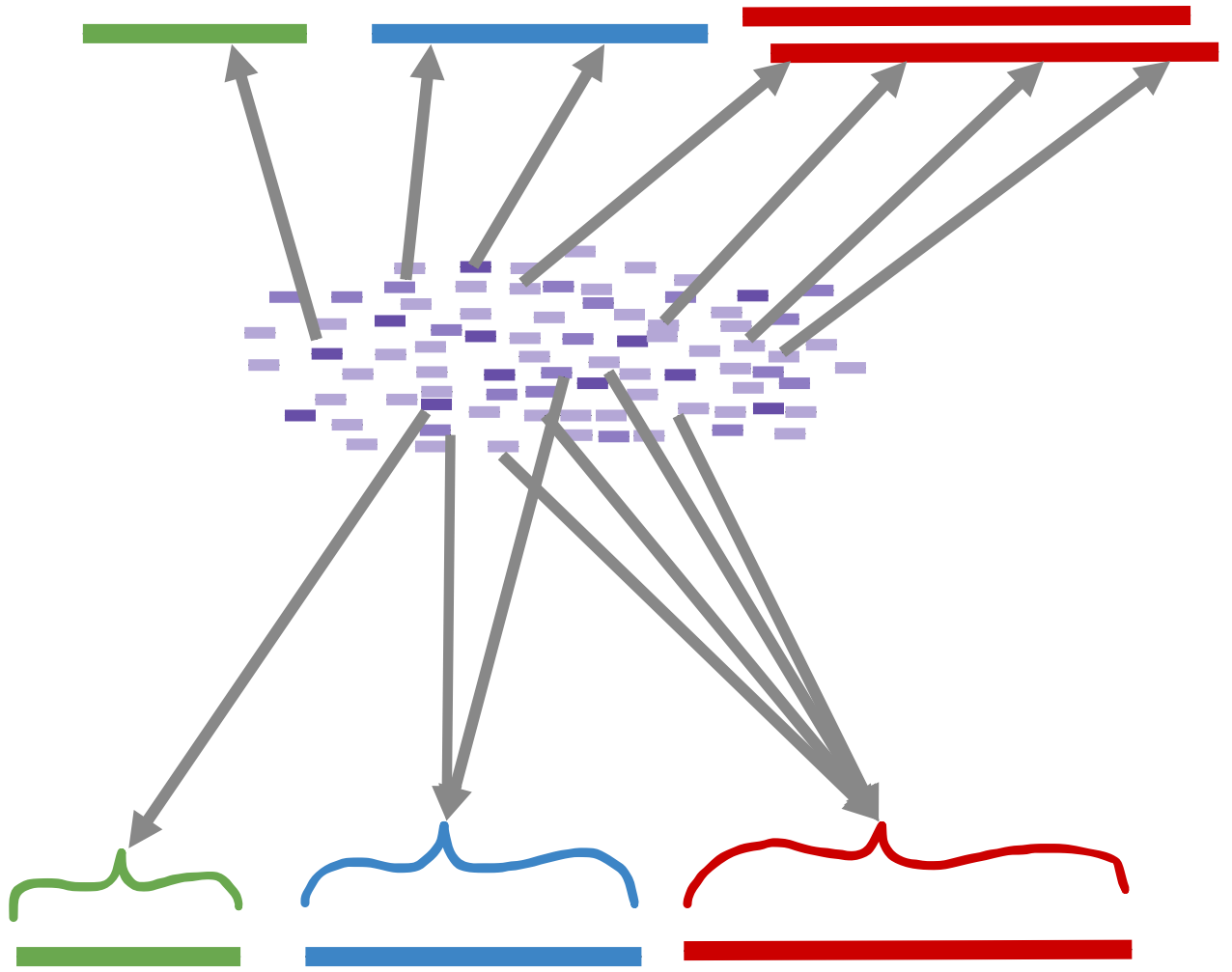
Analogy: Just find books that are compatible and don't worry about which page

Alignment

Specifies where exactly in the transcript this read came from (e.g., at position 478)

Pseudoalignment

Specifies that it came somewhere from this transcript (i.e., compatible)



Bypassing alignment accelerates quantification

Pseudoalignment: This method, used by tools like Kallisto, skips the full alignment process. Instead of mapping each read to a specific position, pseudoalignment identifies which transcripts are compatible with a given read

- **Pros**: Faster and less resource-intensive than alignment-based methods
- **Cons**: It may lack certain details, such as the position and orientation of reads, which are useful for correcting technical biases

After today, you should be able to

1. Discuss the importance of normalization and quantification in RNA-seq data analysis.
2. Explain the relevance of pseudoalignment instead of read mapping.
3. **Understand the purpose of Salmon's generative model.**
4. Describe how salmon handles experimental biases in transcriptomics data.
5. Communicate the principles of inference in Salmon.

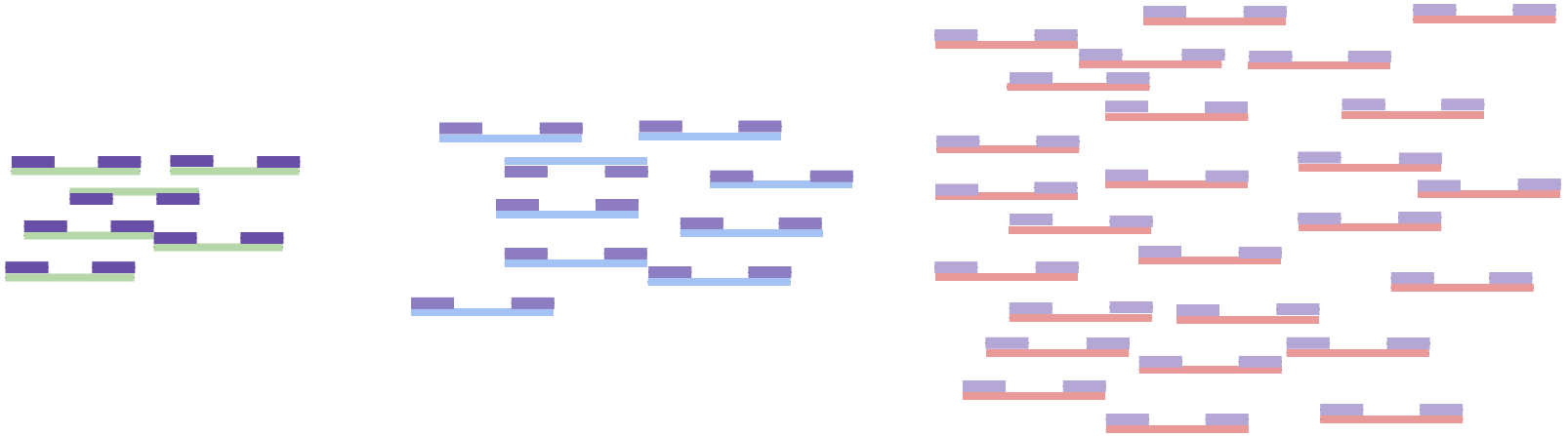
Let's understand our problem

Initial sample

Has some number of transcripts

Fragments

After PCR amplification and fragmentation



Reads

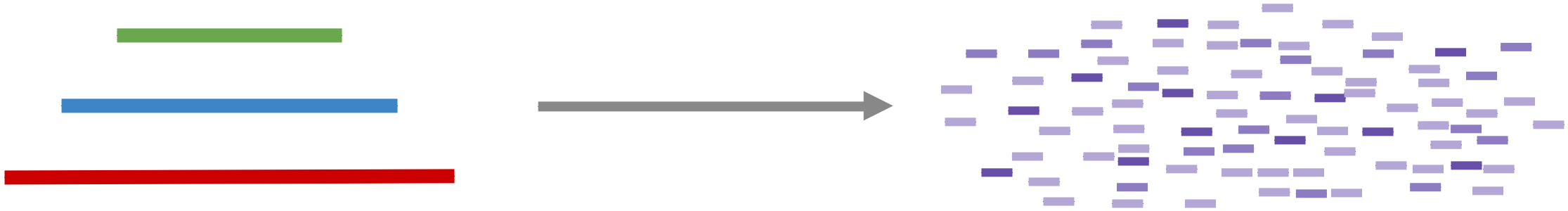
Sequencing with imperfections

We have to use reads to quantity our initial sample

What is a generative model?

Generative model: A statistical model that explains how the observed data are generated from the underlying system

Defines a computational framework that produces sequencing reads from a population of transcripts



First, we have to define our model

Salmon's mathematical definition of a transcriptome

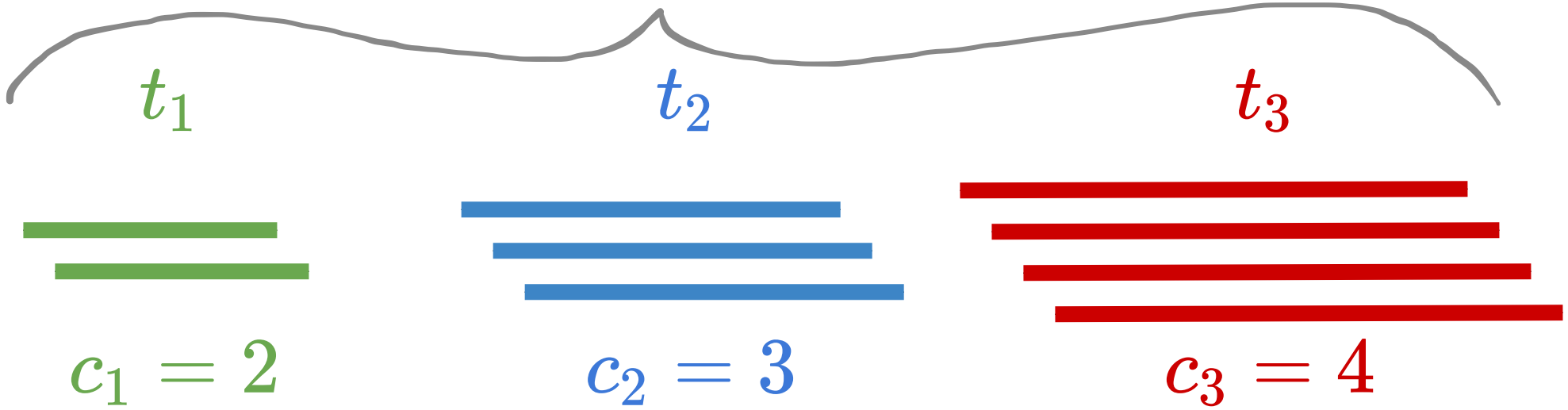
Individual transcripts

Transcript counts

Our whole transcriptome

$$T = \{(t_1, \ldots, t_M), (c_1, \ldots, c_M)\}$$

T $M = 3$

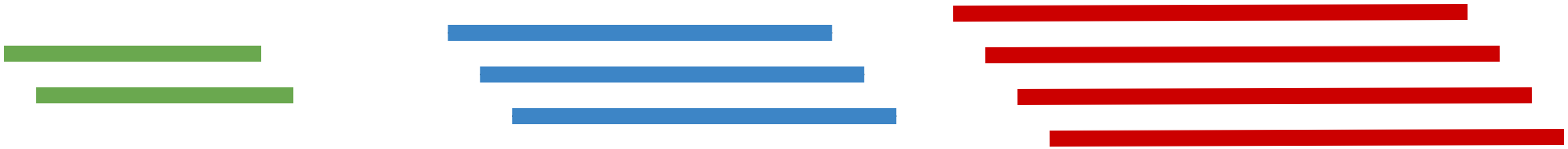


Salmon's formulation of transcript abundance

$c_1 = 2$
$\tilde{l}_1 = 200$

$c_2 = 3$
$\tilde{l}_2 = 300$

$c_3 = 4$
$\tilde{l}_3 = 400$



So far, we have been talking about transcript fractions

$$f_i = \frac{c_i}{\sum_j^M c_j}$$

$$\eta_i = \frac{c_i \tilde{l}_i}{\sum_j^M c_j \tilde{l}_j} \qquad \eta = \begin{bmatrix} \eta_1 \\ \eta_2 \\ \eta_3 \end{bmatrix}$$

We can also take nucleotide fractions by taking into account the effective length of each transcript

This tells us how much of the total RNA pool comes from each transcript

I will explain the effective length later. For now, think of it as a "corrected" length

Converting to relative abundances

τ_i The transcript fraction normalizes nucleotide fraction by the effective length

$$\tau_i = \frac{\frac{\eta_i}{\tilde{l}_i}}{\sum_{j=1}^{M} \frac{\eta_j}{\tilde{l}_j}}$$

Adjusts for the fact that longer transcripts generate more reads

This gives the relative abundance of each transcript *i*

$$\mathrm{TPM}_i = \tau_i \cdot 10^6$$

The **transcript fraction** tells us the proportion of total RNA molecules in the sample that come from transcript *i*

TPM is "Transcripts per million"

Transcript-Fragment Assignment Matrix

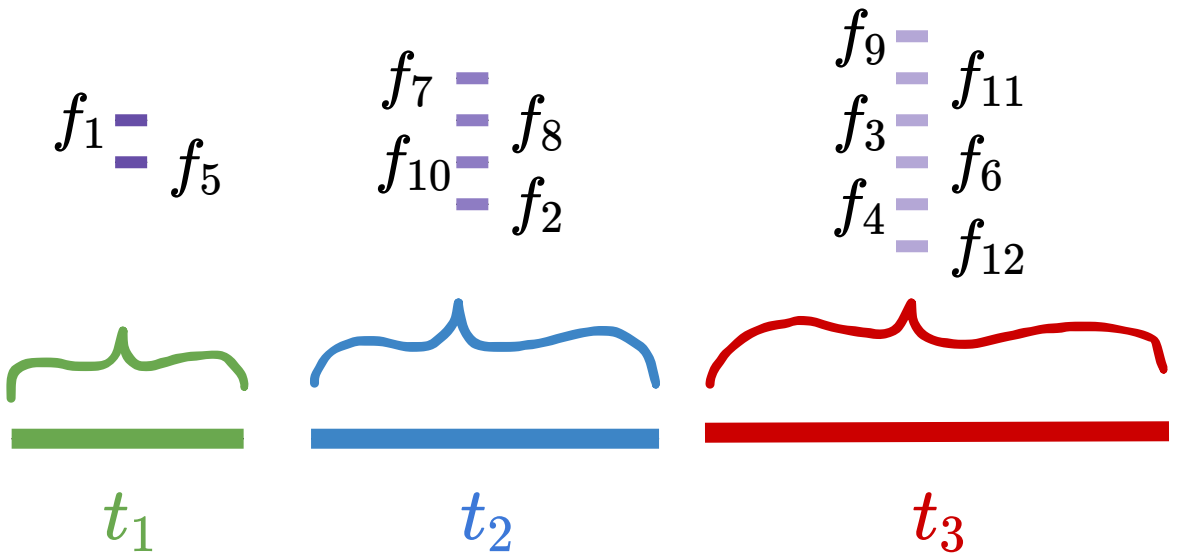
Z is a binary matrix (i.e., all values are 0 or 1)
of *M* transcripts (rows) and *N* fragments (columns)

$$Z = \begin{array}{c} \begin{matrix} \textbf{Fragment 1} & \textbf{Fragment 2} & \textbf{...} & \textbf{Fragment N} \end{matrix} \\ \begin{bmatrix} Z_{11} & Z_{12} & \dots & Z_{1N} \\ Z_{21} & Z_{22} & \dots & Z_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ Z_{M1} & Z_{M2} & \dots & Z_{MN} \end{bmatrix} \end{array} \begin{array}{l} \textbf{Transcript 1} \\ \textbf{Transcript 2} \\ \\ \textbf{...} \\ \\ \textbf{Transcript M} \end{array}$$

$Z_{i,j} = 1$ if fragment *j* is assigned to transcript *i*

Z example

Suppose we have 3 transcripts and 12 fragments

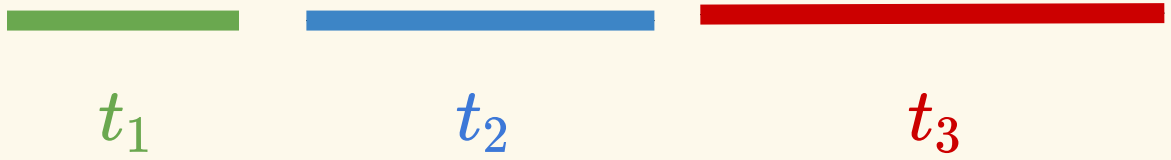


Z is just how we computationally assign fragments to transcripts

$$Z = \begin{array}{c} \begin{matrix} f_1 & f_2 & f_3 & f_4 & f_5 & f_6 & f_7 & f_8 & f_9 & f_{10} & f_{11} & f_{12} \end{matrix} \\ \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix} \end{array} \begin{matrix} t_1 \\ t_2 \\ t_3 \end{matrix}$$

Generative model inference

Known from organism and experiment



Given these inputs, generate a distribution of fragments

Transcript-fragment assignment

$$Z = \begin{bmatrix} Z_{11} & Z_{12} & \dots & Z_{1N} \\ Z_{21} & Z_{22} & \dots & Z_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ Z_{M1} & Z_{M2} & \dots & Z_{MN} \end{bmatrix}$$

Transcript abundance

$$\eta = \begin{bmatrix} \eta_1 \\ \vdots \\ \eta_M \end{bmatrix}$$

N and *M* are same as experiment

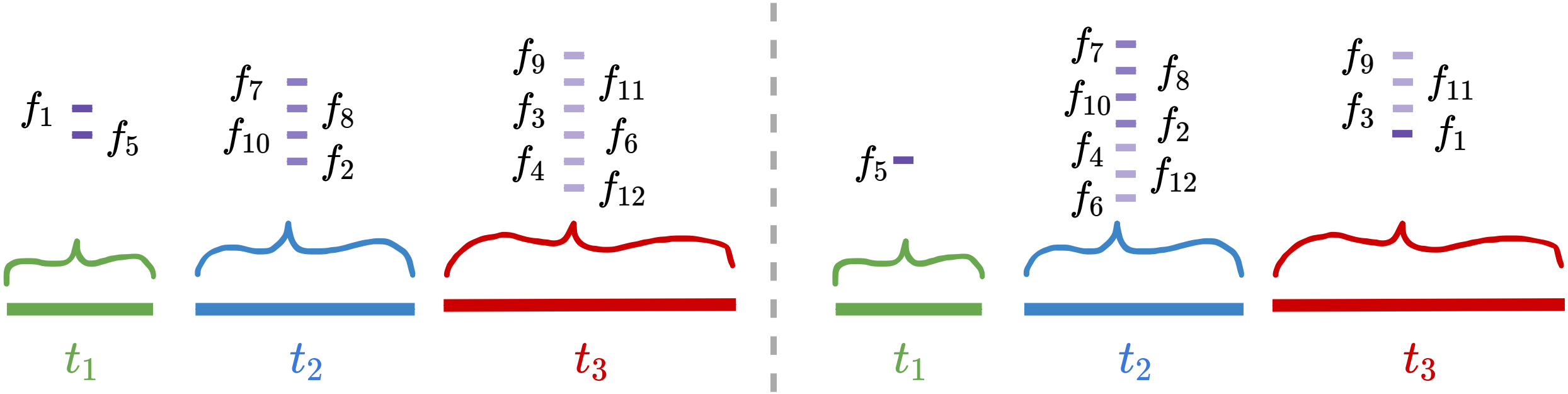
Run 1

Run 2

Probability of observing the sequence fragments

Which scenario is more likely, given our generative model?

We can use probabilistic methods to find parameters that explain our observed distirbution

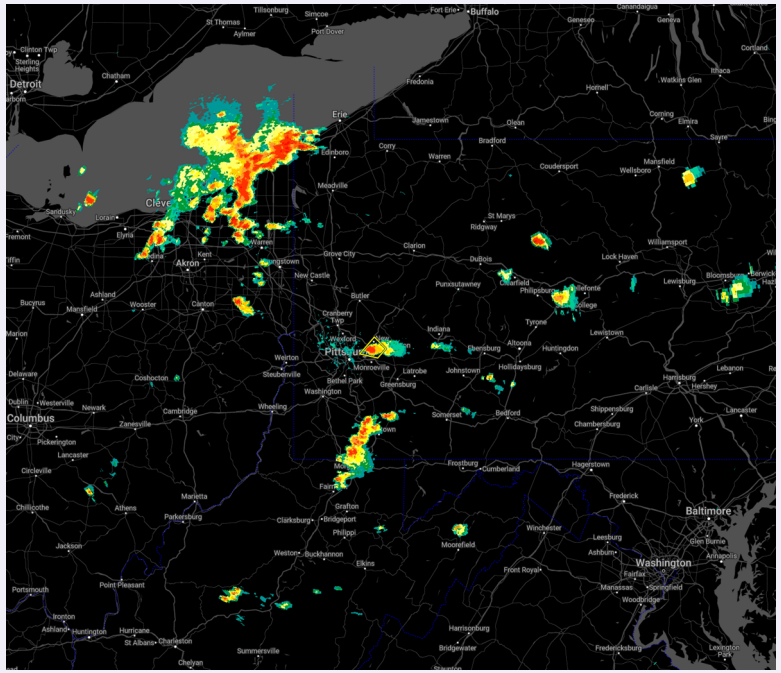


Conditional probability notation

$P\left(a|b\right)$

This reads, "What is the probability of **a** occurring if **b** is true?"

Example



$P\left(\text{Rain}|\text{Radar}\right)$

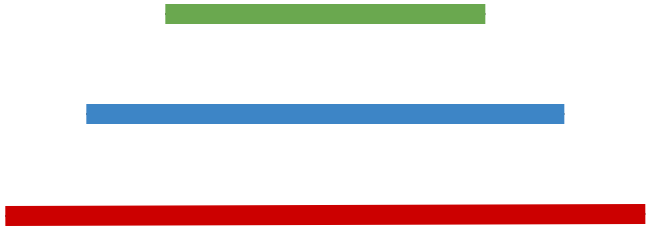
=

Given this **Radar**, what is the probability of **Rain** in Oakland?

Probability of observing the sequenced fragments

$$P\left(F|T,\eta,Z\right)$$

Available transcripts



Transcript-fragment assignment

$$Z = \begin{bmatrix} Z_{11} & Z_{12} & \dots & Z_{1N} \\ Z_{21} & Z_{22} & \dots & Z_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ Z_{M1} & Z_{M2} & \dots & Z_{MN} \end{bmatrix}$$

Transcript abundance

$$\eta = \begin{bmatrix} \eta_1 \\ \vdots \\ \eta_M \end{bmatrix}$$

Given these **parameters**, how probable is it that our experiment generated these observed reads?

Optimize these values until we get the highest probability

After today, you should be able to

1. Discuss the importance of normalization and quantification in RNA-seq data analysis.
2. Explain the relevance of pseudoalignment instead of read mapping.
3. Understand the purpose of Salmon's generative model.
4. **Describe how salmon handles experimental biases in transcriptomics data.**
5. Communicate the principles of inference in Salmon.

Probability of observing the sequenced fragments

We can now compute the probability of observing: Set of fragments F

Given: Transcriptome T

Transcript assignment Z Transcript abundance η

$$P(F|\eta, Z, T) = \prod_{j=1}^{N} \sum_{i=1}^{M} \eta_i P(f_j|t_i)$$

$P(f_j|t_i)$

Probability of observing fragment f_j

given that it comes from transcript t_i

This expression accounts for all possible transcripts a fragment might come from, weighted by how likely that fragment is to come from each transcript

Fragment probabilities

$P(f_j|t_i)$ is a conditional probability that depends on the **position** of the fragment within the transcript, the **length** of the fragment, and any technical biases

In Salmon's quasi-mapping approach, this probability is approximated based on transcript compatibility rather than exact positions.

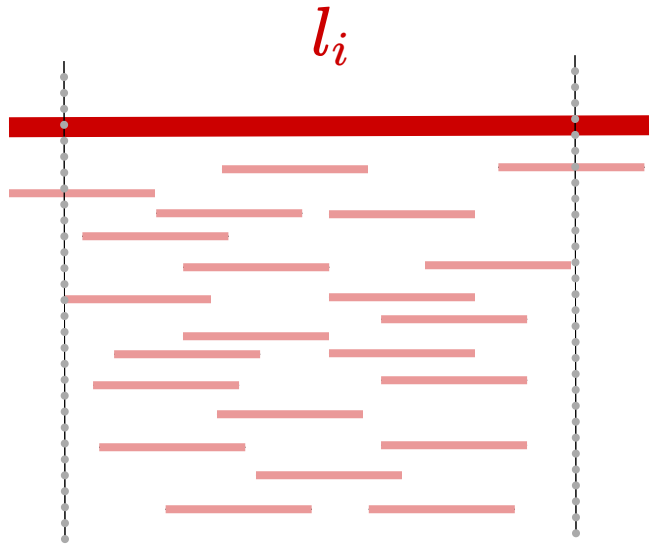
$$P(f_j|t_i) = P(\text{fragment length}, \text{position}, \text{GC content}, \ldots)$$

Positional bias

Fragments that include transcript ends might be too short

Fragments from central regions are more likely to be of optimal length for sequencing reads

A transcript's **effective length** adjusts for the fact that fragments near the ends of a transcript are less likely to be sampled



$$\tilde{l}_i = l_i - \mu_i \qquad \tilde{l}_i < l_i$$

μ_i **Mean of the truncated empirical fragment length distribution**

$$\eta_i = \frac{c_i \tilde{l}_i}{\sum_i c_i \tilde{l}_i}$$

After today, you should be able to

1. Discuss the importance of normalization and quantification in RNA-seq data analysis.
2. Explain the relevance of pseudoalignment instead of read mapping.
3. Understand the purpose of Salmon's generative model.
4. Describe how salmon handles experimental biases in transcriptomics data.
5. **Communicate the principles of inference in Salmon.**

Introduction to inference in Salmon

- **Inference** refers to the process of estimating transcript abundances from observed RNA-seq reads using statistical models.
- Salmon's inference process involves estimating the most likely **abundance** of each transcript that could explain the observed set of fragments (reads).
- It does this by solving a complex, high-dimensional problem where each fragment might map to multiple transcripts.

Two-phase inference in salmon

Salmon processes reads in **two stages**

Online phase

Makes fast, initial estimates of transcript abundances as the reads are processed

Offline phase

Refines these initial estimates using more complex optimization techniques

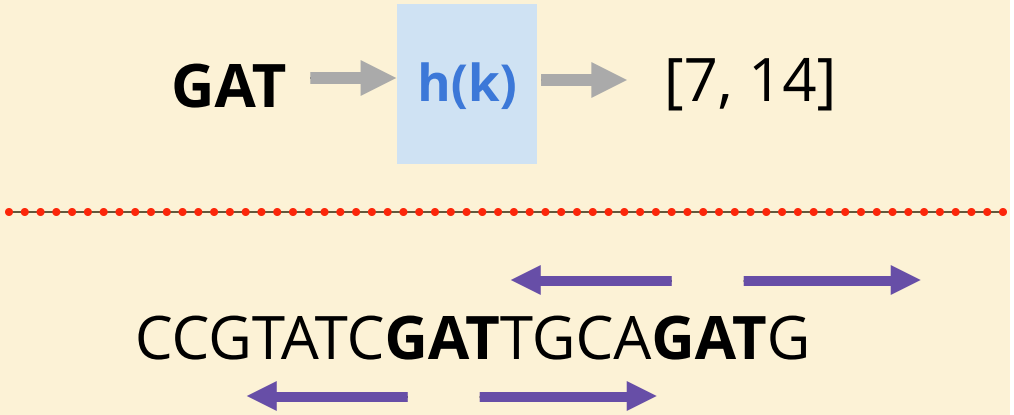
This two-phase approach balances **speed** (in the online phase) with **accuracy** (in the offline phase)

Online phase:
Stochastic variational inference

Initial estimates using quasi-mapping

Quasi-mapping is A fast, lightweight technique used to associate RNA-seq fragments with possible transcripts

Read mapping



Identify seeds, then extend and compute base-by-base alignment

Essentially early stopping of read mapping

Alignment is expensive, so quasi-mapping stops after identify seeds

This is what initializes compatible transcripts and abundance

$$\eta_t \approx \frac{\text{Number of fragments mapping to } t}{\text{Total number of fragments}}$$

Iteratively update parameters based on mini batches

Take current parameters

Mini-batch 1

Compute derivatives from batch

Update parameters (i.e., abundances)

Repeat for each batch

Mini-batch 2

Mini-batch 3

Offline Phase: Expectation-Maximization (EM) algorithm

Offline phase fine tunes transcript abundance

After the online phase, Salmon refines the estimates using a more complex optimization method, typically based on the **Expectation-Maximization (EM) algorithm**

This phase ensures the accuracy of abundance estimates, incorporating the bias corrections learned during the online phase

Likelihood of the Data

The **likelihood** function is central to the inference process in Salmon:

$$\mathcal{L}\{\alpha|F,Z,T\} = \prod_{j=i}^{N}\sum_{i=1}^{M}\hat{\eta}_i Pr\{f_j|t_i\}$$

This is the probability of observing the entire set of fragments F, given the transcriptome T and nucleotide fractions η

Optimize the estimates of α, a vector of the estimated number of reads originating from each transcript

$$\hat{\eta}_i = \frac{\alpha_i}{\sum_j \alpha_j}$$

The goal is to **maximize this likelihood** to infer the most likely values of η, which correspond to the relative abundances of the transcripts

Maximum Likelihood Estimation (MLE)

The goal of **maximum likelihood** is to find the parameters (transcript abundances) that **maximize the probability** of the observed data (sequenced reads)

The **likelihood** function is central to the inference process in Salmon:

$$\mathcal{L}\{\alpha|F,Z,T\} = \prod_{j=i}^{N}\sum_{i=1}^{M}\hat{\eta}_i Pr\{f_j|t_i\}$$

Optimize the estimates of α, a vector of the estimated number of reads originating from each transcript

Given α, η can be directly computed.

Why the EM Algorithm Maximizes the Likelihood

The EM algorithm works by breaking down a difficult problem into two simpler problems:

- In the **E-step**, we estimate the missing information (the assignment of fragments to transcripts) using the current transcript abundance estimates.
- In the **M-step**, we use the estimated assignments to update the transcript abundances, improving the likelihood.

At each iteration, the likelihood of the observed data increases, and the EM algorithm iteratively refines the transcript abundance estimates until it reaches a maximum

Before the next class, you should

Lecture 09:
Quantification

Lecture 10:
Differential gene expression

Today

Thursday

- A04 is due Friday
- Study for exam