Fast Algorithms for Improved Transcriptome Analysis II : Quantification

Rob Patro

CSE 549 Fall 2016

Where might we use quasi-mapping?

We believe there are ***many*** places where this replacement can be made. I'll discuss one in some depth (and mention a second):

1)Transcript-level quantification

- Determine abundance of transcripts from a collection of RNA-seq reads.
- The quasi-mapping information is sufficient to yield estimates *as accurate as full alignment*.

2)*de novo* transcript clustering

- Find groups of related contigs likely from the same transcript / gene
- Such groups help improve downstream analysis (e.g. differential expression testing)

Obviously, alignments are *necessary* for certain types of analysis (e.g. variant detection).

Where might we use quasi-mapping?

We believe there are ***many*** places where this replacement can be made. I'll discuss one in some depth (and mention a second):

1)Transcript-level quantification

- Determine abundance of transcripts from a collection of RNA-seq reads.
- The quasi-mapping information is sufficient to yield estimates *as accurate as full alignment*.

2)*de novo* transcript clustering

- Find groups of related contigs likely from the same transcript / gene
- Such groups help improve downstream analysis (e.g. differential expression testing)

Obviously, alignments are *necessary* for certain types of analysis (e.g. variant detection).

Sequencing Reads

pre-proc.

align to ref.

txp. identification

reference-based

pre-proc.

de novo assembly

align to assembly

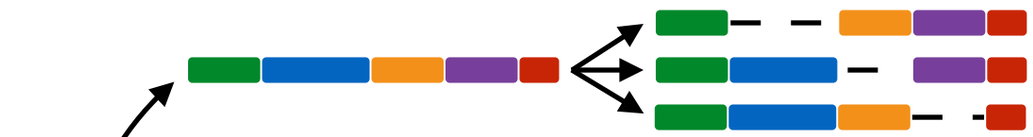
de novo

quantification

DE, Alternative Splicing, etc.

“Higher-level” analysis

Transcript Quantification: An Overview



1 gene ⇒ many variants (isoforms)

Gene 1

Gene M

Sample

10s-100s of millions of
short (35-300 character) “fragments”

Measurement
(RNA-seq)

Inference
(e.g. Salmon)

% Gene 1

% Gene M

isoform A

isoform B

isoform C

Abundance Estimates

Given: (1) Collection of RNA-Seq fragments
(2) A set of known (or assembled) transcript sequences

Estimate: The relative abundance of each transcript

Given: (1) Collection of RNA-Seq fragments
(2) A set of known (or assembled) transcript sequences

Estimate: The relative abundance of each transcript

Question: If we only care about “gene" abundance, can’t we just count the number of reads mapping / aligning to each gene?

Given: (1) Collection of RNA-Seq fragments
(2) A set of known (or assembled) transcript sequences

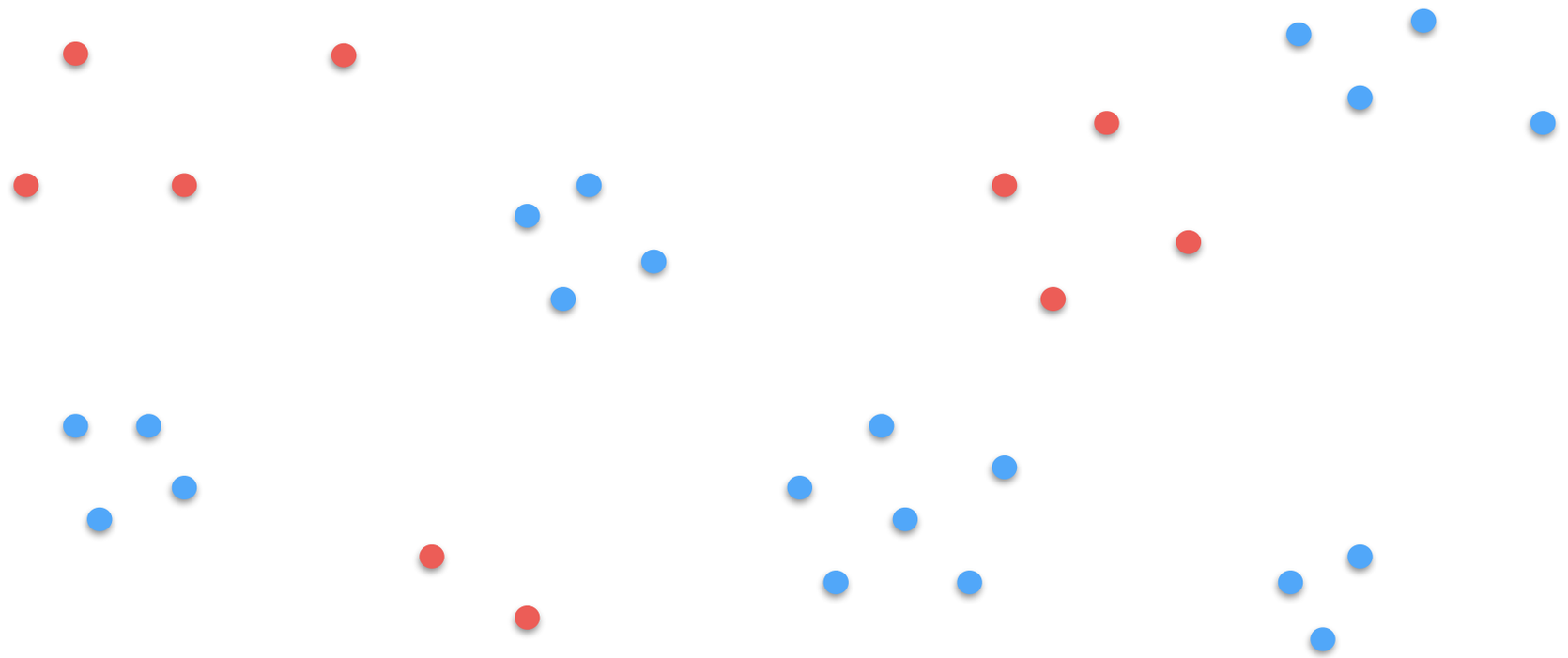
Estimate: The relative abundance of each transcript

Question: If we only care about “gene" abundance, can’t we just count the number of reads mapping / aligning to each gene?

Answer: **No.** I’ll show a general argument (and a few examples) why!

First, consider this non-Biological example

Imagine I have two colors of circle, red and blue. I want to estimate the **fraction of circles** that are red and blue. I'll *sample* from them by tossing down darts.



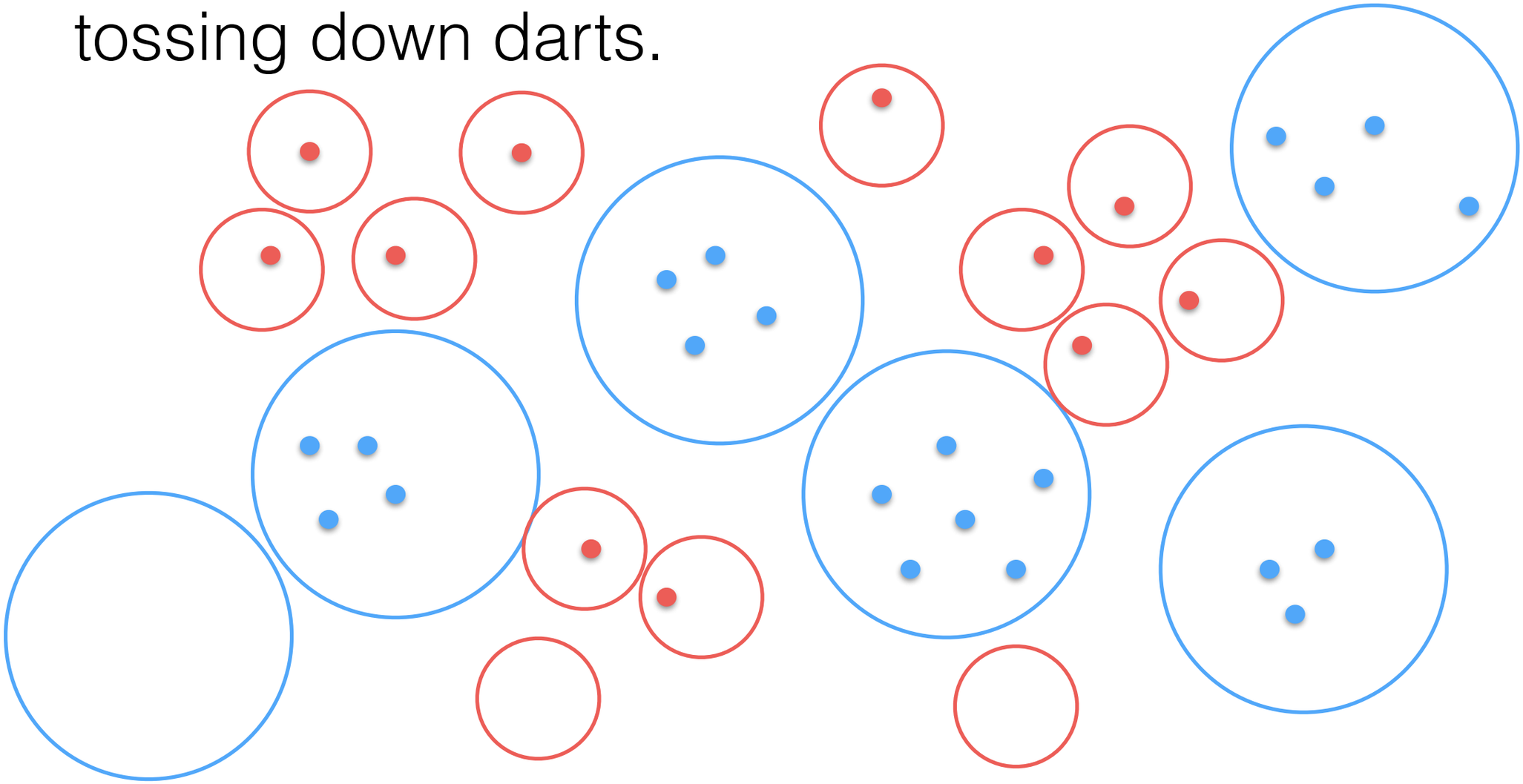
Here, a dot of a color means I hit a circle of that color.

What type of circle is more prevalent?

What is the fraction of red / blue circles?

First, consider this non-Biological example

Imagine I have two colors of circle, red and blue. I want to estimate the **fraction of circles** that are red and blue. I'll *sample* from them by tossing down darts.



You're missing a **crucial piece of information!**

The areas!

First, consider this non-Biological example

Imagine I have two colors of circle, red and blue. I want to estimate the **fraction of circles** that are red and blue. I'll *sample* from them by tossing down darts.

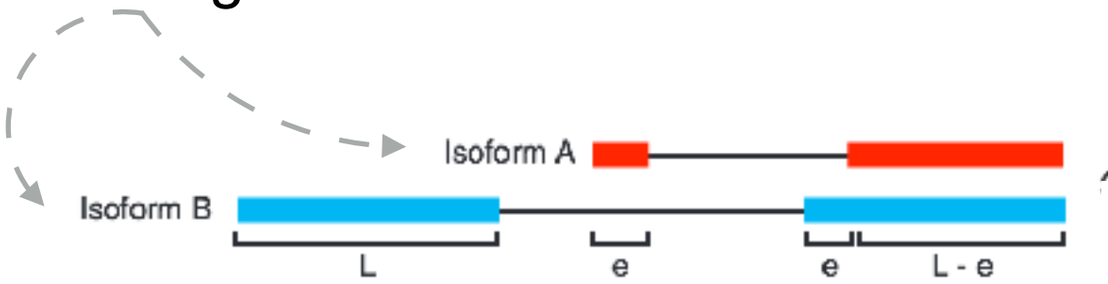
You're missing a **crucial piece of information!**

The areas!

There is an analog in RNA-seq, one needs to know the **length** of the target from which one is drawing to meaningfully assess abundance!

Resolving multi-mapping is fundamental to quantification

Isoform A is half
as long as isoform B



Condition 1	Condition 2	union-model fold-change		true fold-change
		$\log_2\left(\frac{10}{10}\right) = 0$	$0 < 0.32$	$\log_2\left(\frac{\frac{10}{L}}{\frac{6}{L}+\frac{4}{2L}}\right) = 0.32$
		$\log_2\left(\frac{6}{8}\right) = -0.41$	$-0.41 < 0.58$	$\log_2\left(\frac{6/L}{8/2L}\right) = 0.58$
		$\log_2\left(\frac{5}{10}\right) = -1$	$-1 < 0$	$\log_2\left(\frac{\frac{5}{L}}{\frac{10}{2L}}\right) = 0$

Key point : The length of the *actual molecule* from which the fragments derive is crucially important to obtaining accurate abundance estimates.

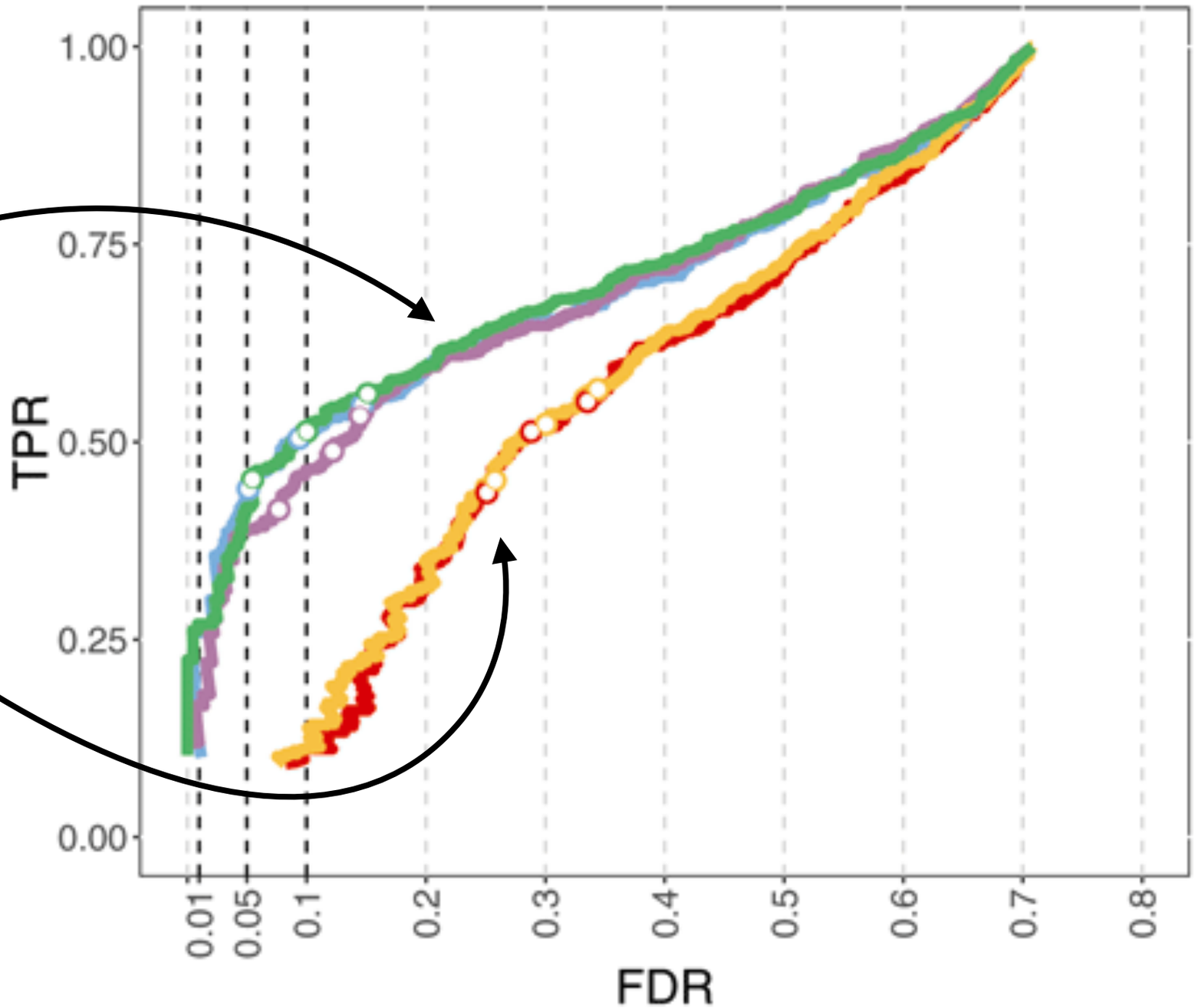
Adapted from: Trapnell, Cole, et al. "Differential analysis of gene regulation at transcript resolution with RNA-seq." Nature biotechnology 31.1 (2013): 46-53.

Resolving multi-mapping is fundamental to quantification

These errors can affect DGE calls

Variants of Salmon

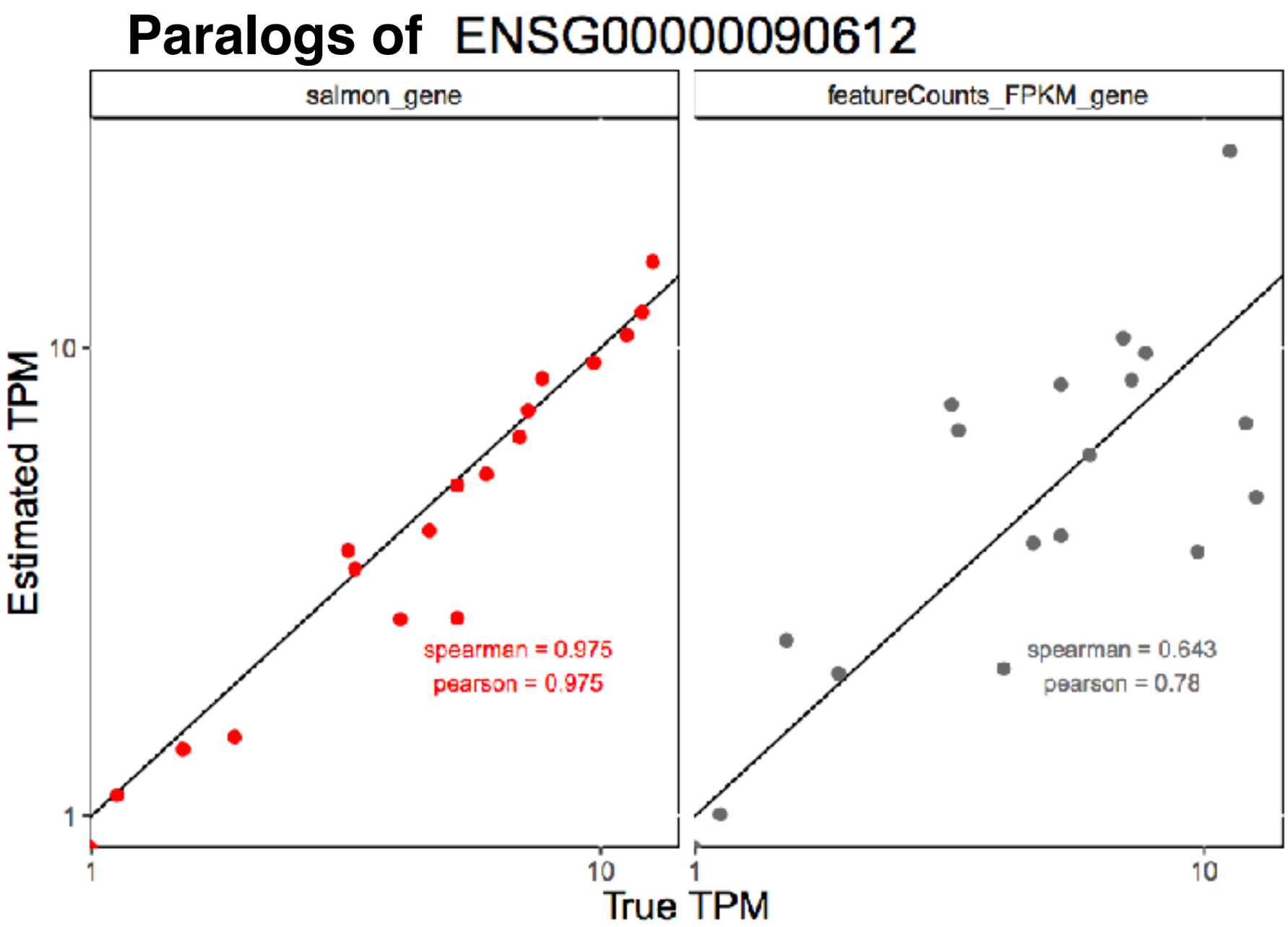
Variants of “counting”



From: Soneson C, Love MI and Robinson MD 2016 [version 2; referees: 2 approved] F1000Research 2016, 4:1521 (doi: 10.12688/f1000research.7563.2)

Resolving multi-mapping is fundamental to quantification

Can even affect abundance estimation in **absence** of alternative-splicing (e.g. paralogous genes)

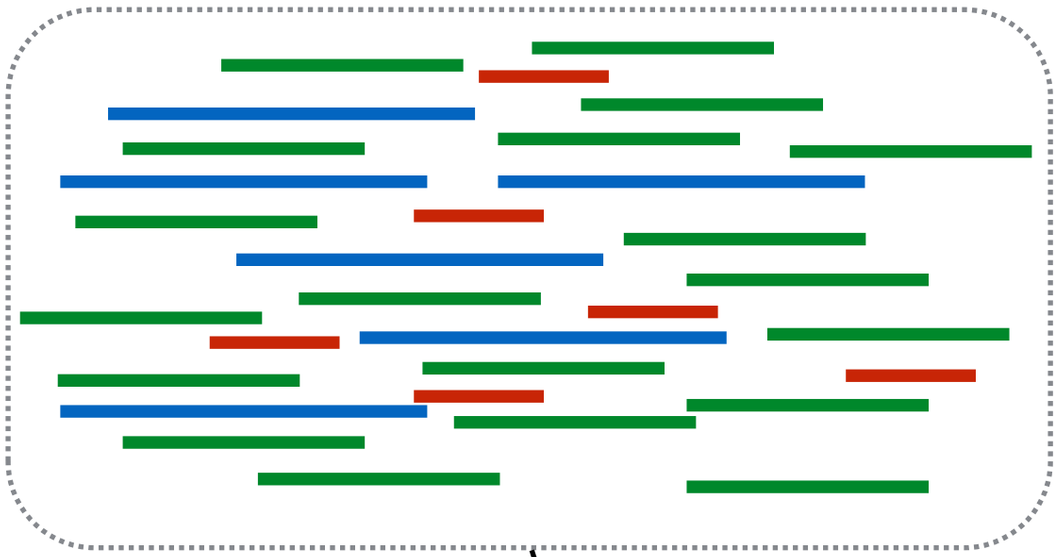


From: Soneson C, Love MI and Robinson MD 2016 [version 2; referees: 2 approved] F1000Research 2016, 4:1521 (doi: 10.12688/f1000research.7563.2)

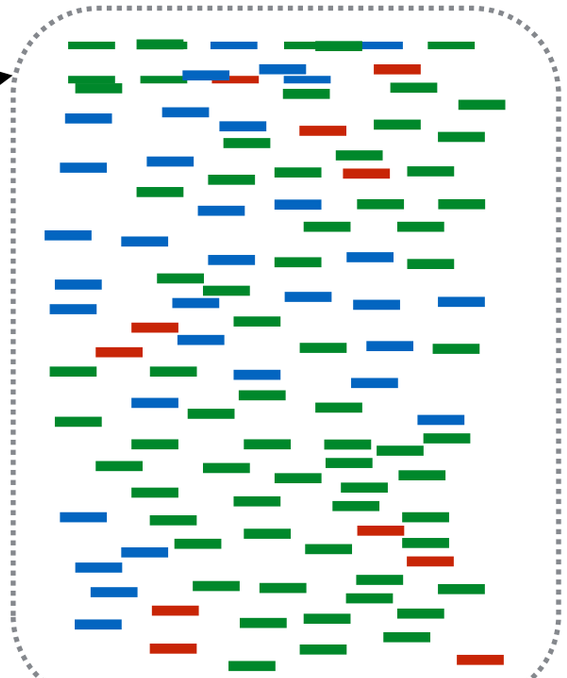
How do we do something better than “counting”?

Think about the “ideal” RNA-seq experiment . . .

Experimental Mixture



Read set



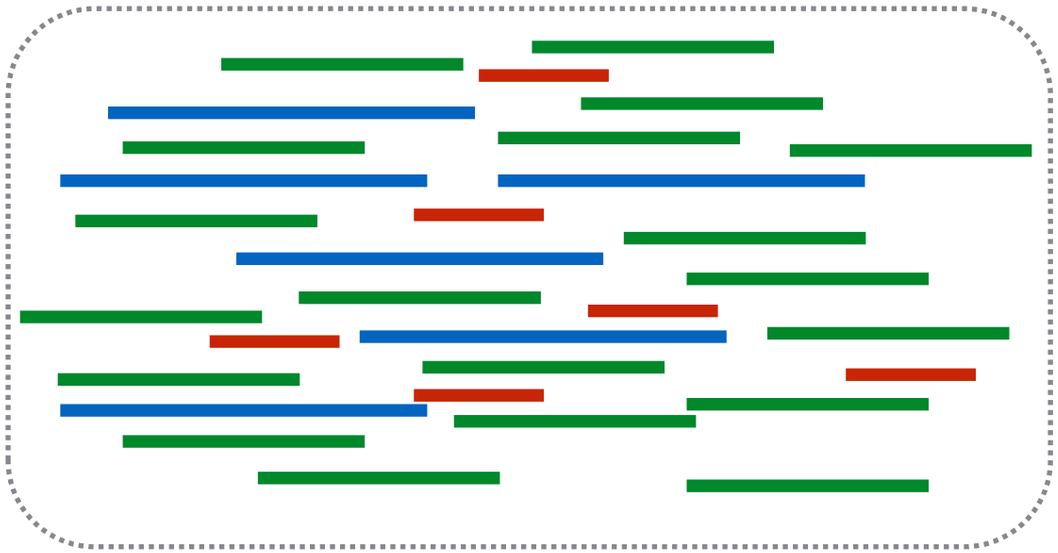
sequencing *oracle*

Pick a transcript **t** ∝ count * length

Pick a position **p** on **t** uniformly “at random”

How do we do something better than “counting”?

Experimental Mixture



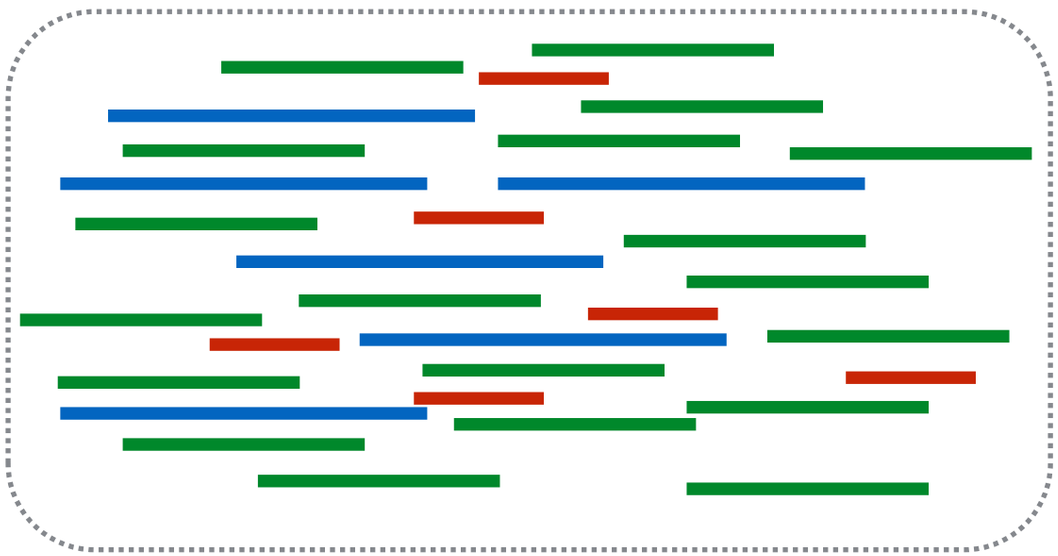
length( ) = 100 x 6 copies = 600 nt ~ 30% blue

length( ) = 66 x 19 copies = 1254 nt ~ 60% green

length( ) = 33 x 6 copies = 198 nt ~ 10% red

How do we do something better than “counting”?

Experimental Mixture



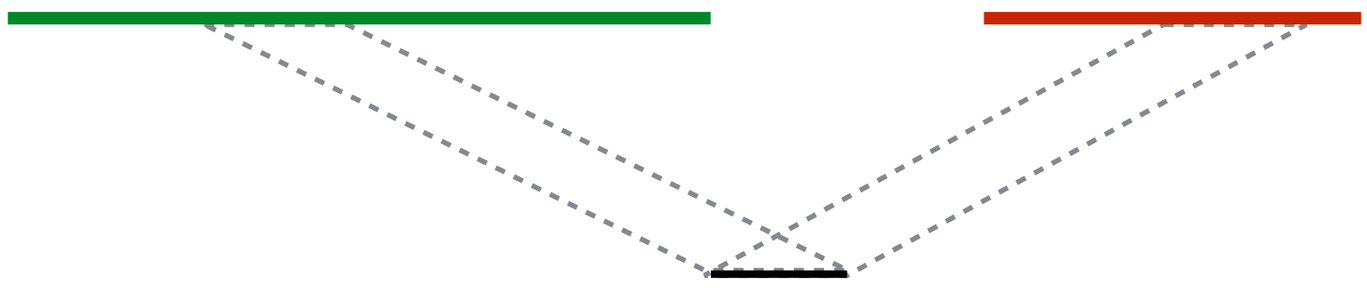
length( ) = 100 x 6 copies = 600 nt ~ 30% blue

length( ) = 66 x 19 copies = 1254 nt ~ 60% green

length( ) = 33 x 6 copies = 198 nt ~ 10% red

We call these values η = [0.3, 0.6, 0.1] the nucleotide fractions, they become the primary quantity of interest

Resolving a single multi-mapping read



Say we *knew* the η, and observed a read that mapped ambiguously, as shown above. What is the probability that it truly originated from G or R?

$$\Pr\{r \text{ from } G\} = \frac{\frac{\eta_G}{\text{length}(G)}}{\frac{\eta_G}{\text{length}(G)} + \frac{\eta_R}{\text{length}(R)}} = \frac{\frac{0.6}{66}}{\frac{0.6}{66} + \frac{0.1}{33}} = 0.75$$

$$\Pr\{r \text{ from } R\} = \frac{\frac{\eta_R}{\text{length}(R)}}{\frac{\eta_G}{\text{length}(G)} + \frac{\eta_R}{\text{length}(R)}} = \frac{\frac{0.1}{33}}{\frac{0.6}{66} + \frac{0.1}{33}} = 0.25$$

normalization factor

length( ) = 100 x 6 copies = 600 nt ~ 30% blue

length( ) = 66 x 19 copies = 1254 nt ~ 60% green

length( ) = 33 x 6 copies = 198 nt ~ 10% red

So how do we estimate abundance “correctly”?

Key idea: Find the set of transcript abundances that maximizes the probability of the observed data — this is done by *probabilistic* assignment of fragments to transcripts.

That is: We’re asking for the maximum likelihood estimates of transcript abundance

$$\arg\max_{\boldsymbol{\rho}\in\mathbf{P}} \mathcal{L}(\boldsymbol{\rho}; x_1, \ldots, x_n)$$

abundances — parameters of a generative model

observations — alignments of reads to transcripts

So how do we estimate abundance “correctly”?

Finding the maximum likelihood estimates first requires defining the likelihood:

We’ll define it in terms of parameters alpha

$$\alpha_t \quad := \quad \mathbb{P}(f \in t) \quad = \quad \frac{\rho_t \tilde{l}_t}{\sum_{r \in T} \rho_r \tilde{l}_r}$$

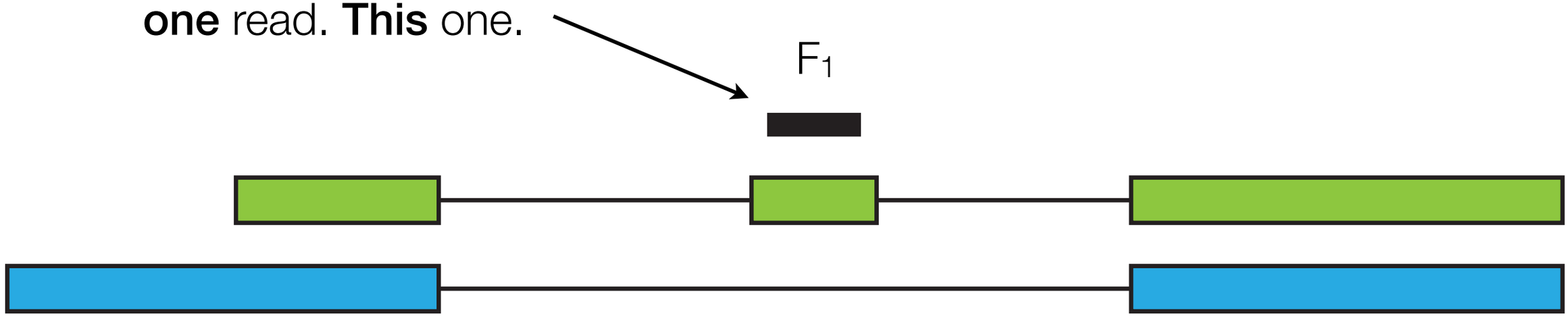
which are relatable, directly, to the rhos

$$\rho_t \quad = \quad \frac{\frac{\alpha_t}{\tilde{l}_t}}{\sum_{r \in T} \frac{\alpha_r}{\tilde{l}_r}}$$

*Pachter, Lior. "Models for transcript quantification from RNA-Seq." arXiv preprint arXiv:1104.3889 (2011).

Defining the likelihood function

Suppose we sequenced just **one** read. **This** one.



A few things need to happen to get this read as opposed to all the others we could have gotten:

We need to pick out a transcript from the RNA pool that could generate this read:

$$\text{Prob(Picking the green transcript)} = \frac{\text{copies of the green transcript}}{\text{total number of transcripts in the pool}}$$

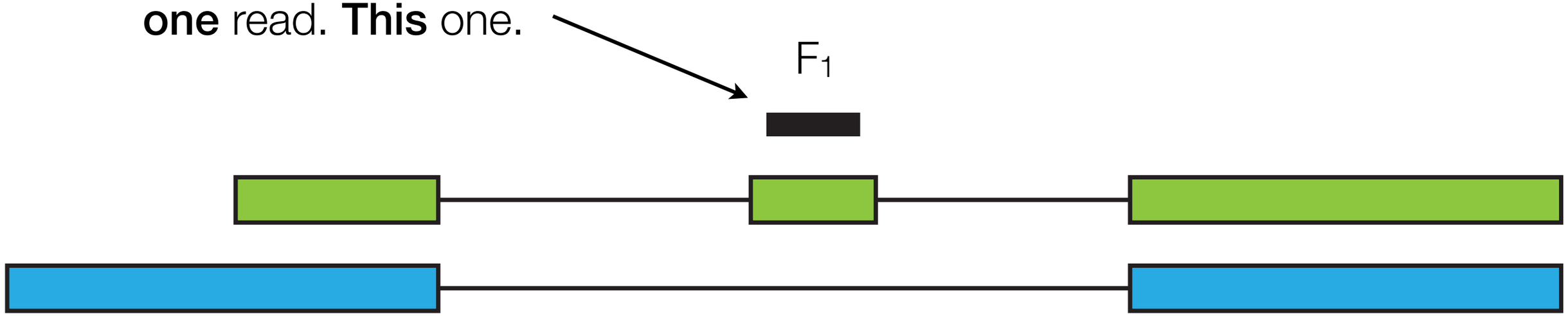
Then, we need to pick this read from that transcript over all the others.

$$\text{Prob(picking this read)} = \frac{1}{\text{length of green transcript}}$$

slide from Cole Trapnell (http://www.stat115.org/lectures/stat115_rnaseq.pdf)

Defining the likelihood function

Suppose we sequenced just **one** read. **This** one.



not normalized by length

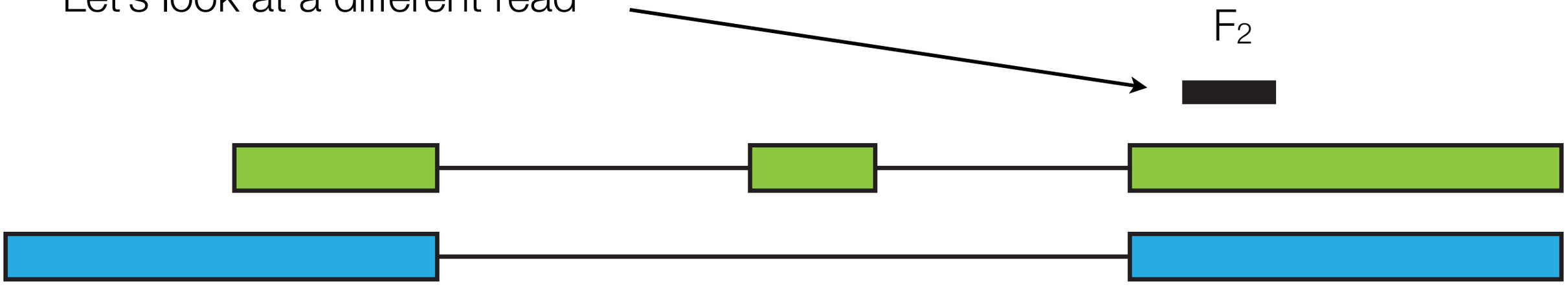
So given a relative abundance for the green transcript, which we'll call α_{green} we can calculate the probability of getting F_1.

$$\Pr(F_1 \in T_{\text{green}}) = \Pr(F_1 \mid \alpha_{\text{green}}) = \frac{\alpha_{\text{green}}}{\ell_{\text{green}}}$$

slide from Cole Trapnell (http://www.stat115.org/lectures/stat115_rnaseq.pdf)

Defining the likelihood function

Let's look at a different read



F_2 could have come from either transcript, so we have to consider two ways of getting it:

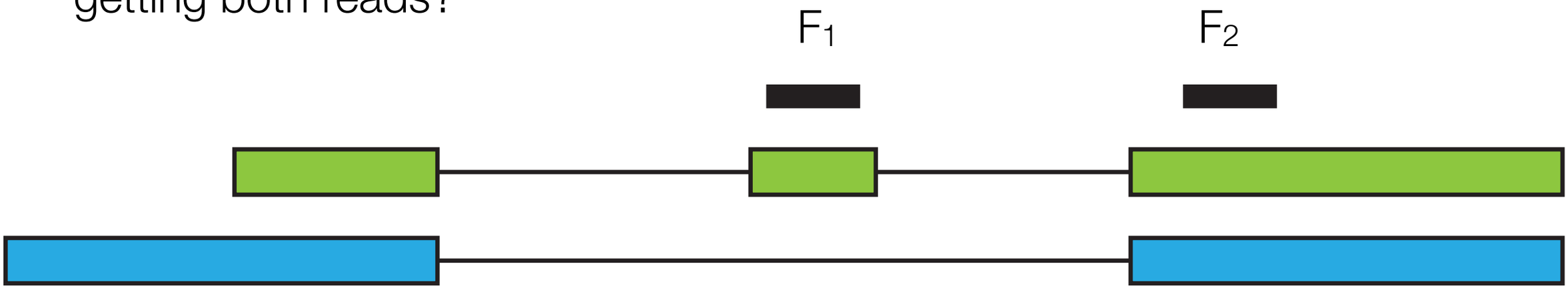
$$\Pr(F_2 \in T_{\text{green}} \text{ or } F_2 \in T_{\text{blue}}) = \Pr(F_2 \mid \alpha) = \frac{\alpha_{\text{green}}}{\ell_{\text{green}}} + \frac{\alpha_{\text{blue}}}{\ell_{\text{blue}}}$$

That is, in order to know the probability of getting F_2, we need to know the abundances of both the transcripts it might have come from.

slide from Cole Trapnell (http://www.stat115.org/lectures/stat115_rnaseq.pdf)

Defining the likelihood function

What are the chances of getting both reads?



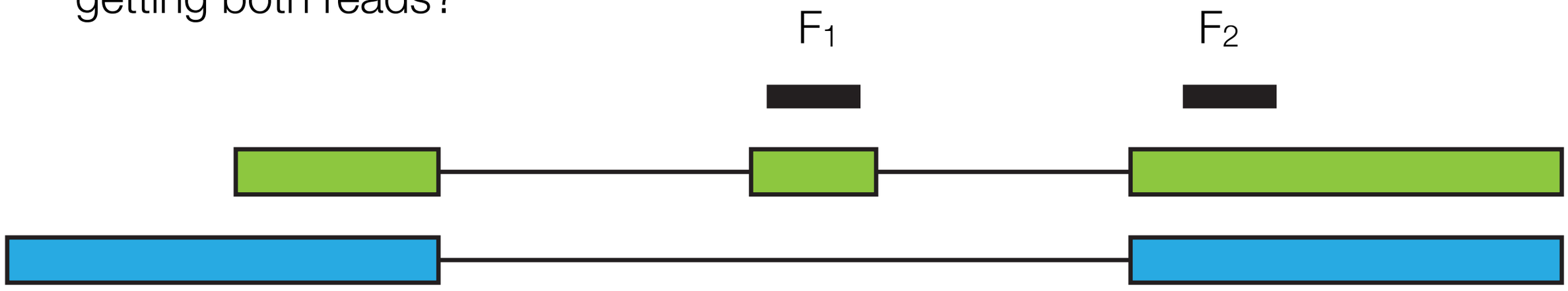
To get both F_1 and F_2, we just need to multiply the two probabilities!

$$\Pr(F_1 \in T_{\text{green}} \text{ and } F_2 \in T_{\text{green}} \text{ or } F_2 \in T_{\text{blue}}) = \Pr(F \mid \alpha) = \left(\frac{\alpha_{\text{green}}}{\ell_{\text{green}}}\right) \cdot \left(\frac{\alpha_{\text{green}}}{\ell_{\text{green}}} + \frac{\alpha_{\text{blue}}}{\ell_{\text{blue}}}\right)$$

slide from Cole Trapnell (http://www.stat115.org/lectures/stat115_rnaseq.pdf)

Defining the likelihood function

What are the chances of getting both reads?



Let's look at this probability as a *function* of alpha:

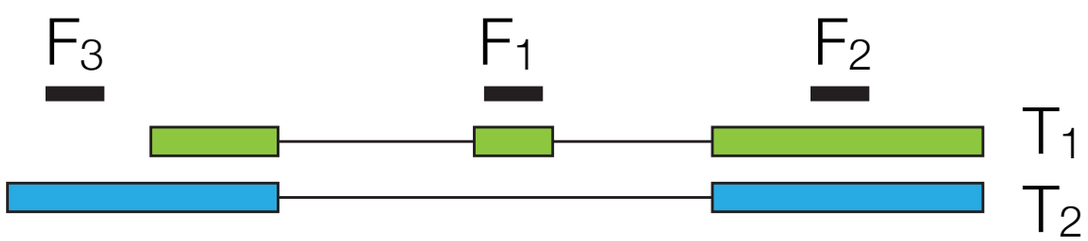
$$\mathcal{L}(\alpha; F) = \mathcal{L}(\alpha) = \left(\frac{\alpha_{\text{green}}}{\ell_{\text{green}}}\right) \cdot \left(\frac{\alpha_{\text{green}}}{\ell_{\text{green}}} + \frac{\alpha_{\text{blue}}}{\ell_{\text{blue}}}\right)$$

Given a input assignment of abundances to transcripts (the alphas), this function returns a number. The greater the number, the better the chances of seeing the reads we actually see.

slide from Cole Trapnell (http://www.stat115.org/lectures/stat115_rnaseq.pdf)

Defining the likelihood function

We can take any set of reads and any set of transcripts, and build one of these likelihood functions:



Compatibility table

	F_1	F_2	F_3
T_1	✓	✓	✗
T_2	✗	✓	✓

$$\mathcal{L}(\alpha; F) = \mathcal{L}(\alpha) = \left(\frac{\alpha_{\text{green}}}{\ell_{\text{green}}}\right) \cdot \left(\frac{\alpha_{\text{green}}}{\ell_{\text{green}}} + \frac{\alpha_{\text{blue}}}{\ell_{\text{blue}}}\right) \cdot \left(\frac{\alpha_{\text{blue}}}{\ell_{\text{blue}}}\right)$$

Now we want to find the values of alpha that **maximize** this likelihood function.

slide from Cole Trapnell (http://www.stat115.org/lectures/stat115_rnaseq.pdf)

Likelihood Function

With the simplest generative model, we get a likelihood function that looks like this:

$$\mathcal{L}(\alpha) = \prod_{t \in T} \left(\frac{\alpha_t}{\tilde{l}_t} \right)^{X_t}$$

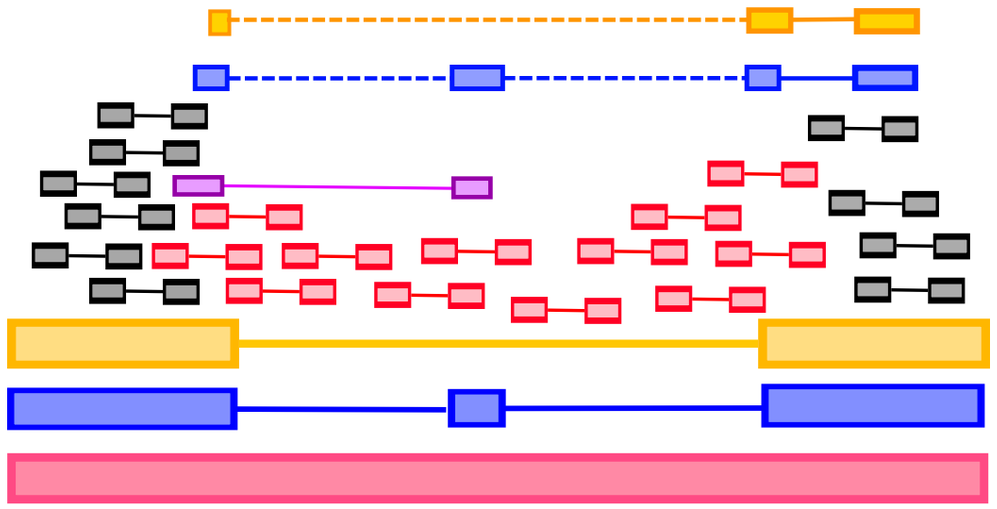
$$\propto \prod_{t \in T} \alpha_t^{X_t}$$

X_t: fragments compatible w/ transcript t

*Pachter, Lior. "Models for transcript quantification from RNA-Seq." arXiv preprint arXiv:1104.3889 (2011).

Assigning reads to isoforms

Problem: infer which transcript each fragment came from



Some fragments could have come from any transcript (black), while others only one (blue, yellow). The purple fragment could have come from either the red or the blue one.

Conditional probability that a fragment came from a given isoform is a function of that isoform's abundance!

slide from Cole Trapnell (http://www.stat115.org/lectures/stat115_rnaseq.pdf)

Finding the MLE

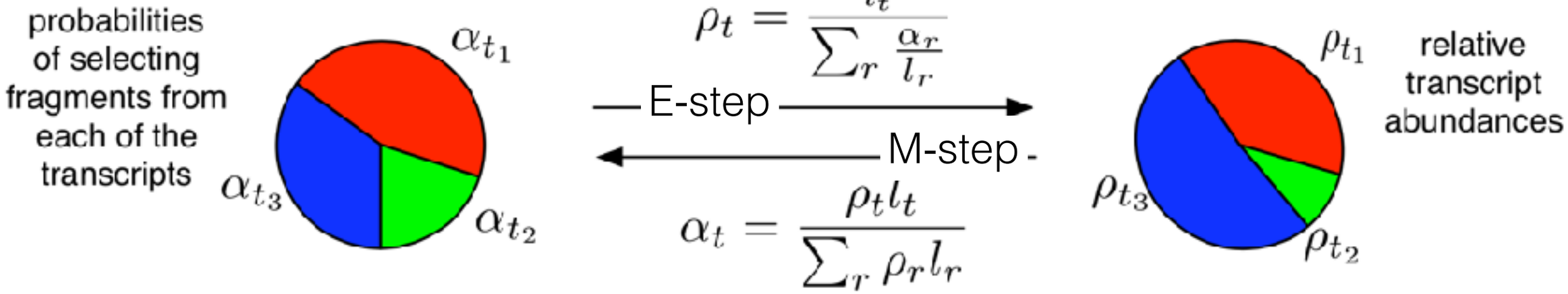
This problem lends itself very well to an Expectation Maximization (EM) approach.

Essentially:

While not converged:

E-step	Assign fragments to transcripts (probabilistically) using current estimates of transcript abundance.
M-step	Re-estimate transcript abundance using probabilistic fragment assignments.

The EM steps, visually



*Pachter, Lior. "Models for transcript quantification from RNA-Seq." arXiv preprint arXiv:1104.3889 (2011).

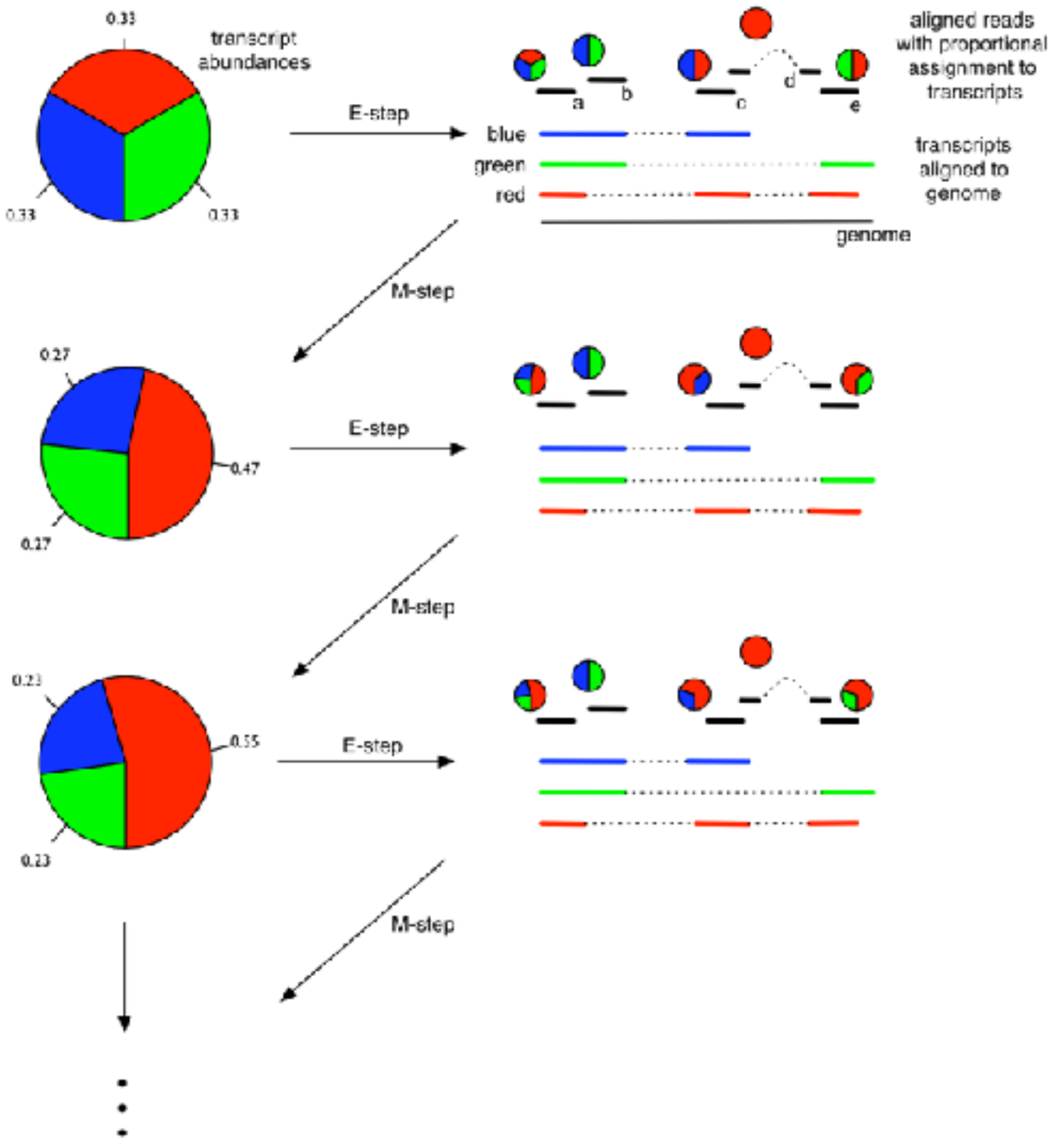
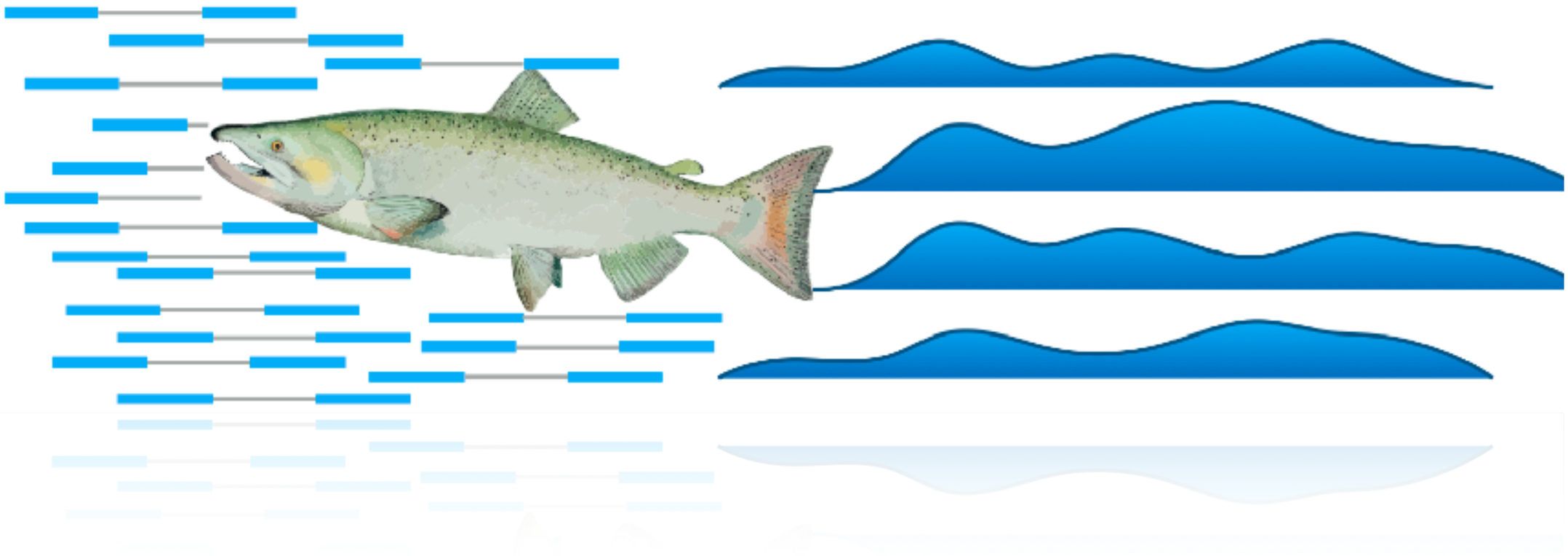


FIGURE 4. Illustration of the EM algorithm. The gene has three isoforms (`red`, `green`, `blue`) of the same length. There are five reads (a,b,c,d,e) mapping to the gene. One maps to all three isoforms, one only to red, and the other three to each of the three pairs of isoforms. Initially every isoform is assigned the same abundance $(\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$. During the *expectation* (E) step reads are proportionately assigned to transcripts according to the isoform abundances. Next, during the *maximization* (M) step isoform abundances are recalculated from the proportionately assigned read counts. Thus, for example, the abundance of `red` after the first M step is estimated by $0.47 = (0.33 + 0.5 + 1 + 0.5)/(2.33 + 1.33 + 1.33)$.

*Pachter, Lior. (2011)

Transcript Quantification

Salmon provides accurate, fast, and bias-aware transcript expression estimates using dual-phase inference



Official website: http://combine-lab.github.io/salmon/

GitHub repository: https://github.com/COMBINE-lab/salmon

joint work with Geet Duggal, Mike Love, Rafael Irizarry & Carl Kingsford

A probabilistic view of RNA-Seq quantification

nucleotide fractions

known transcriptome

assumes independence of fragments

$$\Pr\{\mathcal{F} \mid \boldsymbol{\eta}, \mathcal{T}\} = \prod_{j=1}^{N} \Pr\{f_j \mid \boldsymbol{\eta}, \mathcal{T}\}$$

observed fragments (reads)

We can safely truncate Pr{ti | **η**} to 0 for transcripts where a fragment doesn't map.

$$= \prod_{j=1}^{N} \sum_{i=1}^{M} \Pr\{t_i \mid \boldsymbol{\eta}\} \cdot \Pr\{f_j \mid t_i, \boldsymbol{z}_{ji} = 1\}$$

Prob. of selecting t_i *given* **η**

Depends on abundance estimate

Prob. of generating fragment f_j *given* it originates from t_i

Independent of abundance estimate

We want to find the values of **η** that ***maximize*** this probability.
We can do this (at least locally) using the EM algorithm.

*Li, Bo, and Colin N. Dewey. "RSEM: accurate transcript quantification from RNA-Seq data with or without a reference genome." BMC bioinformatics 12.1 (2011): 1.

A probabilistic view of RNA-Seq quantification

a fragment
starting at given position

$$\Pr\{f_j \mid t_i\} = \Pr\{\ell \mid t_i\} \cdot \Pr\{p \mid t_i, \ell\} \cdot \Pr\{o \mid t_i\} \cdot \Pr\{a \mid f_j, t_i, p, o, \ell\}$$

a fragment
of the given length

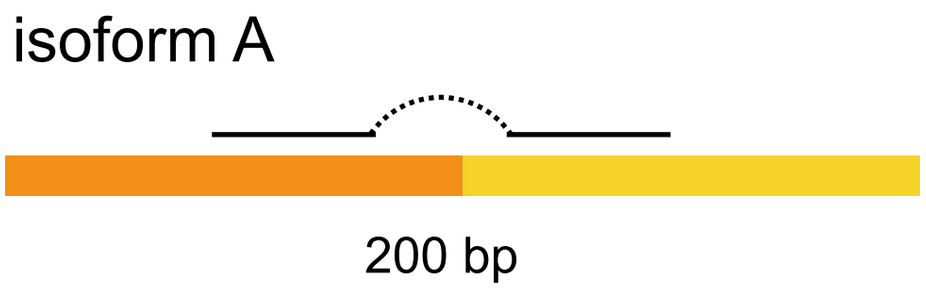
a fragment
of given orientation

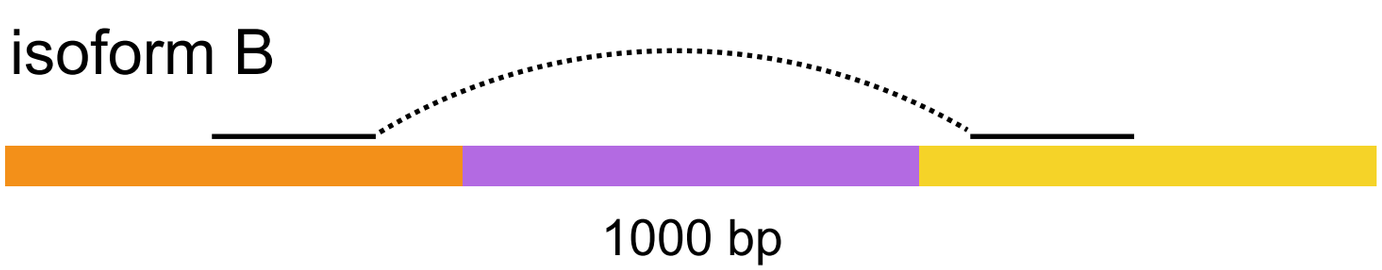
generating the given
alignment / mapping

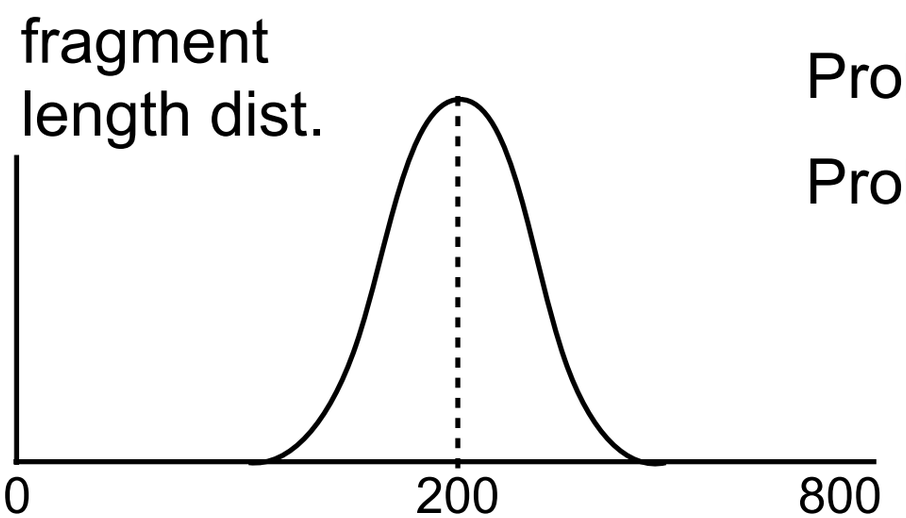
- Salmon estimates an auxiliary model *from the data* for each term (e.g. fragment length, fragment start position, etc.)
- Accounts for sample-specific parameters and biases.
- Also includes modeling of e.g. seq-specific and GC-fragment bias not shown in above equation.

Why does $\Pr\{f_j \mid t_i\}$ matter?

Consider the following scenario:





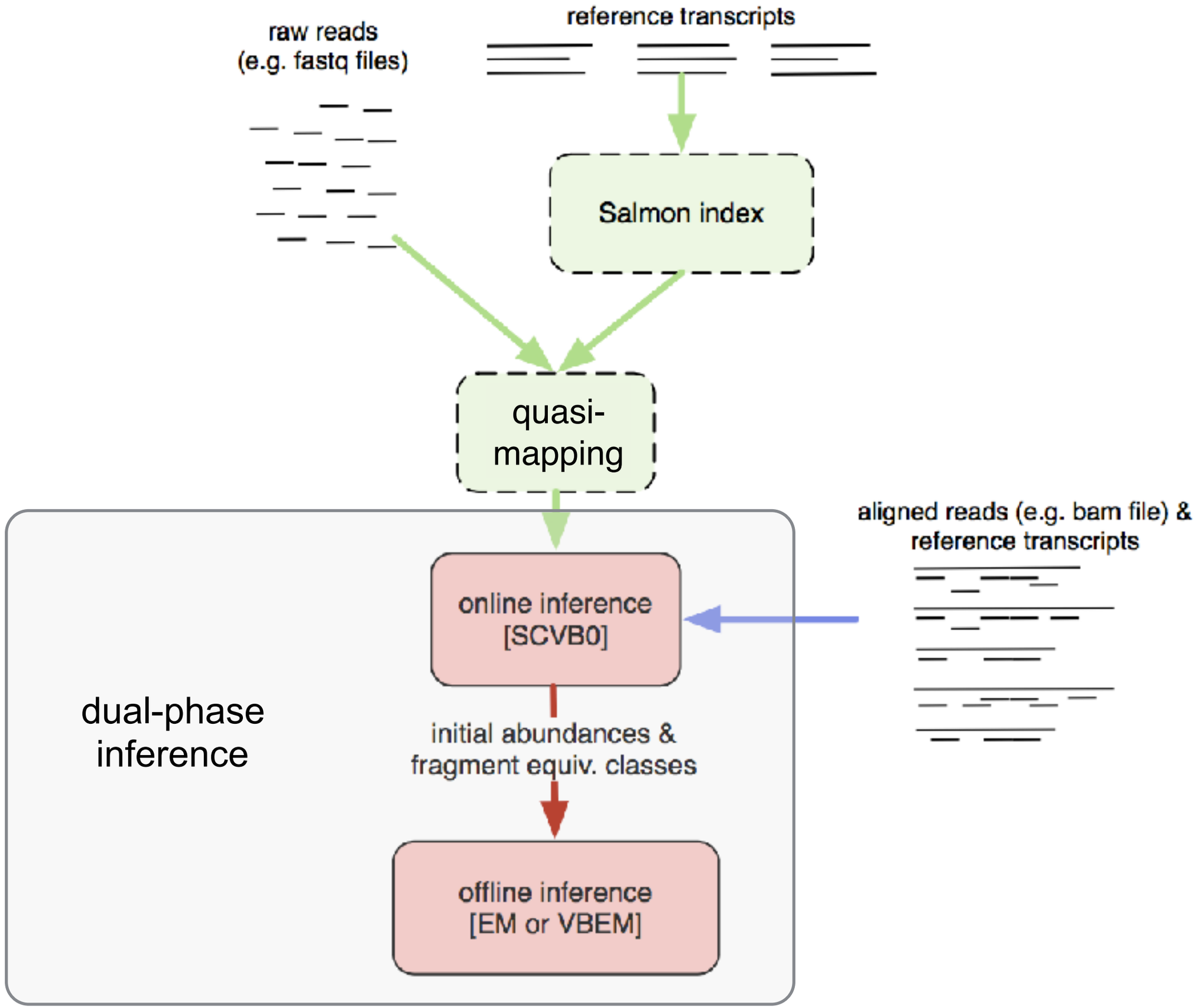


Aux. model provides *strong* information about origin of a fragment!

Prob of observing a fragment of size ~200 is **large**

Prob of observing a fragment of size ~1000 is **very small**

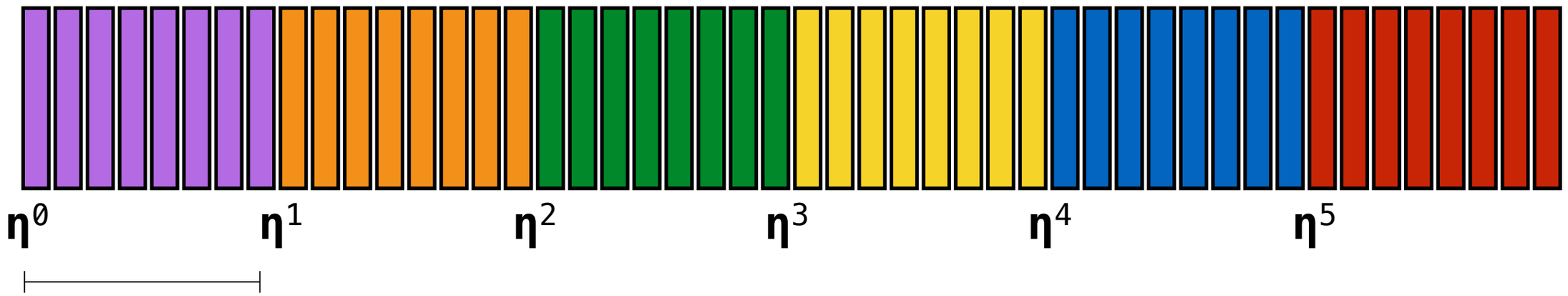
Salmon's "pipeline"



Phase 1: Online Inference

online inference [SCVB0] *

Process fragments in batches:



Compute local $\boldsymbol{\eta}'$ using $\boldsymbol{\eta}^{t-1}$ & current "bias" model to allocate fragments

Update global nucleotide fractions: $\boldsymbol{\eta}^t = \boldsymbol{\eta}^{t-1} + a^t \boldsymbol{\eta}'$

Weighting factor that decays over time

Update "bias" model

Place mappings in **equivalence classes**

- Have access to *all fragment-level information* when making these updates
- Often converges very quickly.
- Compare-And-Swap (CAS) for synchronizing updates of different batches

* Based on: Foulds et al. Stochastic collapsed variational Bayesian inference for latent Dirichlet allocation. ACM SIGKDD, 2013.

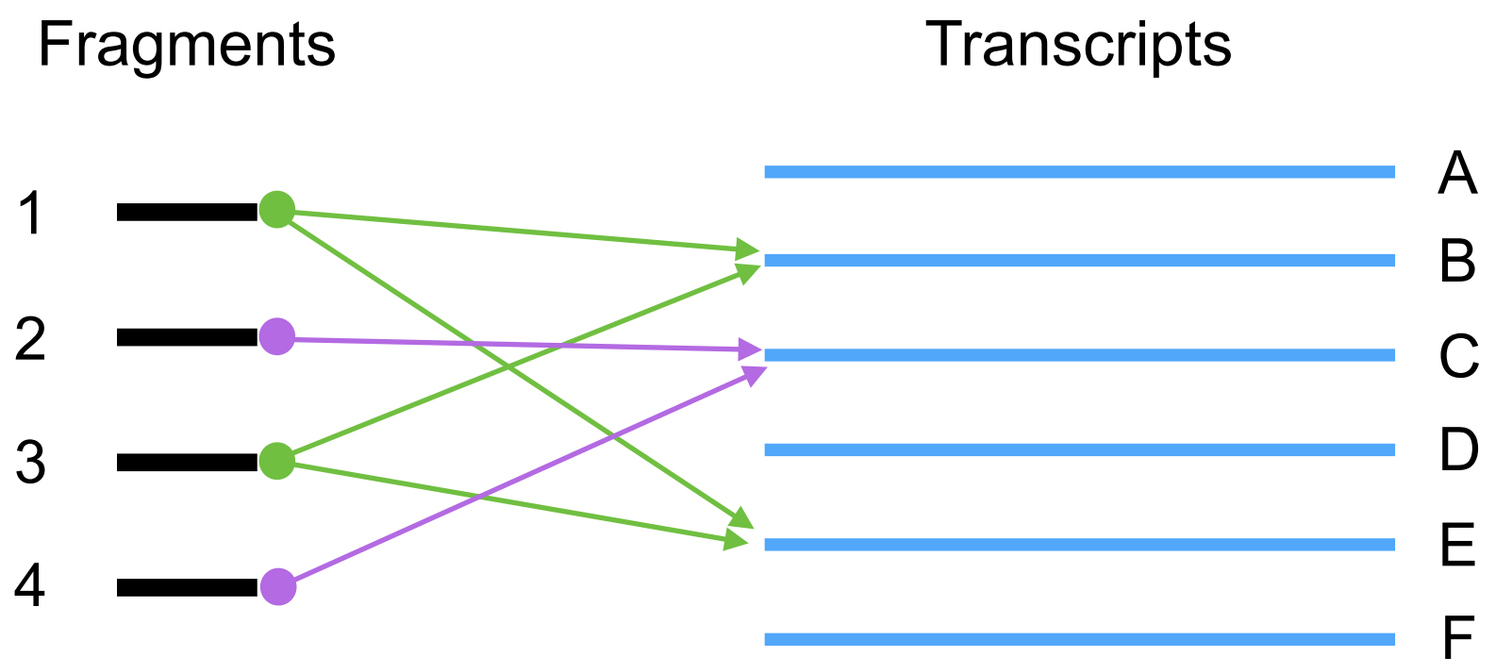
Give each transcript *appropriate* prior mass η^0 (init.)

```
For each mini-batch B^t of reads {

 For each read r in B^t {

  For each alignment a of r {
   compute (un-normalized) prob of a using η^(t-1), and aux params
  }
  normalize alignment probs & update local transcript weights η'
  add / update the equivalence class for read r
  sample a ∈ r to update auxiliary models
 }
 update global transcript weights given local transcript
 weights according to "update rule" ⟹ η^t = η^(t-1) + w^t η'
}
```

mini-batches processed in parallel by different threads

additive nature of updates mitigates effects of no synchronization between mini-batches

Fragment Equivalence Classes

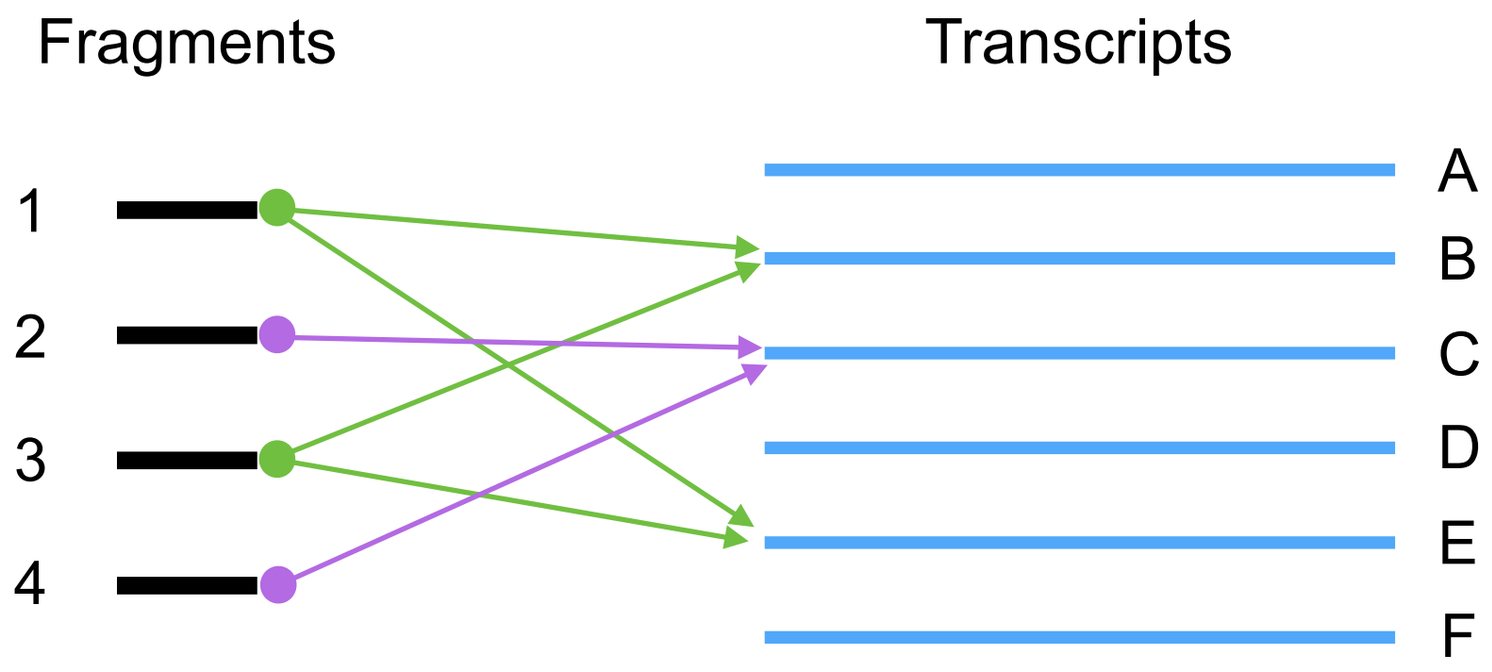


Reads 1 & 3 both map to transcripts B & E
Reads 2 & 4 both map to transcript C

We have 4 reads, but only 2 eq. classes of reads

eq. Label	Count	Aux weights
{B,E}	2	$w^{\{B,E\}}_B, w^{\{B,E\}}_E$
{C}	2	$w^{\{C\}}_C$

Fragment Equivalence Classes



Reads 1 & 3 both map to transcripts B & E
Reads 2 & 4 both map to transcript C

w^j_i encodes the "affinity" of class *j* to transcript *i* according to the model. This is $P\{f_j \mid t_i\}$, aggregated for all fragments in a class.

We have 4 reads, but only 2 eq. classes of reads

eq. Label	Count	Aux weights
{B,E}	2	$w^{\{B,E\}}_B, w^{\{B,E\}}_E$
{C}	2	$w^{\{C\}}_C$

Exploring even "richer" notions of equivalence classes.

Equivalence classes in RNA-Seq

Long history of this idea — collapsing “redundant” reads

Salzman et al.[1] — equiv classes defined on exons / exon-pairs implied by multimapping reads (requires annotation).

Nicolae et al.[2] — equiv classes defined on txps, implied by multimapping reads *require proportional* conditional probabilities

Turro et al.[3] — equiv classes defined on txps, implied by multimapping reads

Patro et al.[4] — equiv classes defined on txps, implied by shared k-mers

Bray et al.[5] — equiv classes defined on txps, implied by shared T-DBG contigs & multimapping reads

1:Salzman, Julia, Hui Jiang, and Wing Hung Wong. "Statistical modeling of RNA-Seq data." Statistical science: a review journal of the Institute of Mathematical Statistics 26.1 (2011).

2:Nicolae, Marius, et al. "Estimation of alternative splicing isoform frequencies from RNA-Seq data." Algorithms for Molecular Biology 6.1 (2011): 1.

3:Turro, Ernest, et al. "Haplotype and isoform specific expression estimation using multi-mapping RNA-seq reads." Genome biology 12.2 (2011): 1.

4:Patro, Rob, Stephen M. Mount, and Carl Kingsford. "Sailfish enables alignment-free isoform quantification from RNA-seq reads using lightweight algorithms." Nature biotechnology 32.5 (2014): 462-464.

5:Bray, Nicolas L., et al. "Near-optimal probabilistic RNA-seq quantification." Nature biotechnology 34.5 (2016): 525-527.

The number of equivalence classes is small

	Yeast	Human	Chicken
# contigs	7353	107,389	335,377
# samples	6	6	8
Total (paired-end) reads	~36,000,000	~116,000,000	~181,402,780
Avg # eq. classes (across samples)	5197	100,535	222,216

The **# of equivalence classes grows with the complexity of the transcriptome** — independent of the # of sequence fragments.

Typically, ***two or more orders of magnitude*** fewer equivalence classes than sequenced fragments.

The offline **inference** algorithm **scales in # of fragment equivalence classes**.

Optimizing the objective

offline inference
[EM or VBEM]

Consider our likelihood

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

where α_i is prop. to # of reads assigned to t_i

our ML objective can be re-written **in terms of our *eq. classes***

$$\mathcal{L}\{\boldsymbol{\alpha} \mid \mathcal{F}, \boldsymbol{Z}, \mathcal{T}\} = \prod_{\mathcal{C}^j \in \mathcal{C}} \left(\sum_{t_i \in \boldsymbol{t}^j} \hat{\eta}_i w_i^j \right)^{d^j}$$

count of eq. class j

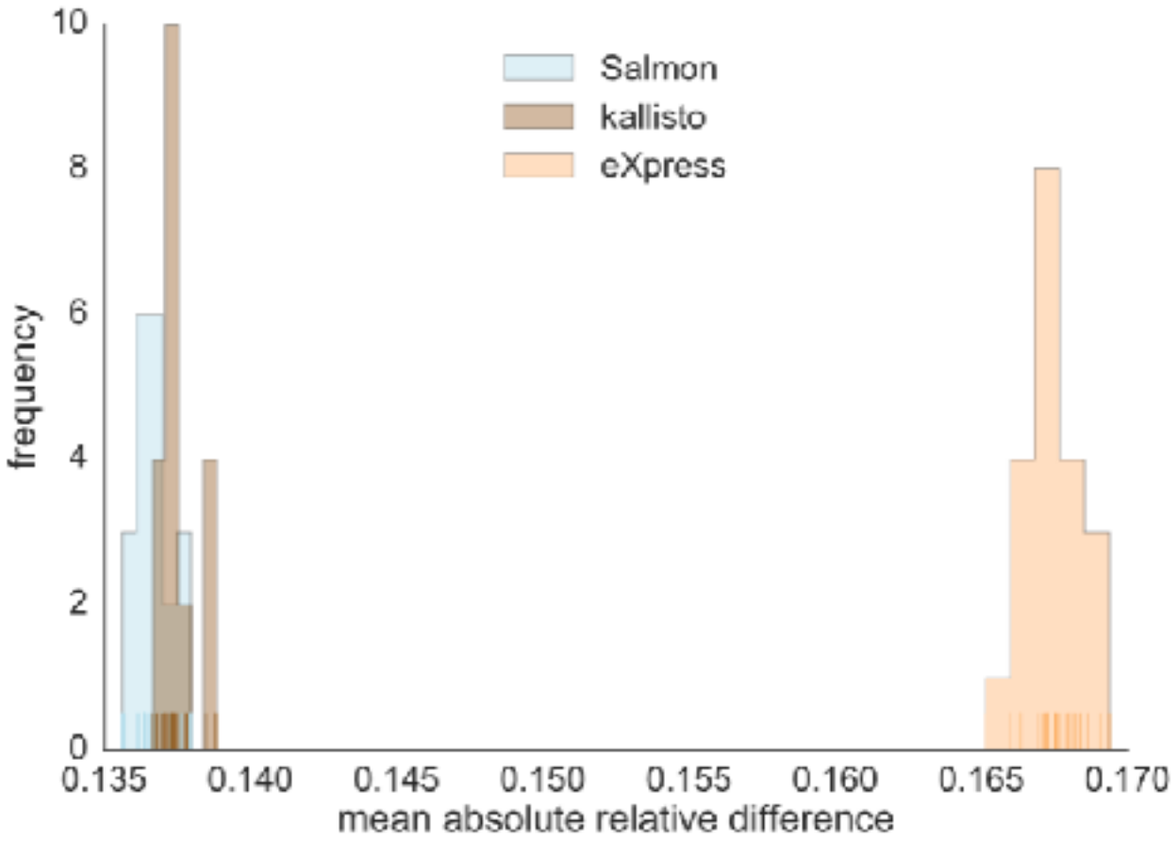
weight of t_i in eq. class j

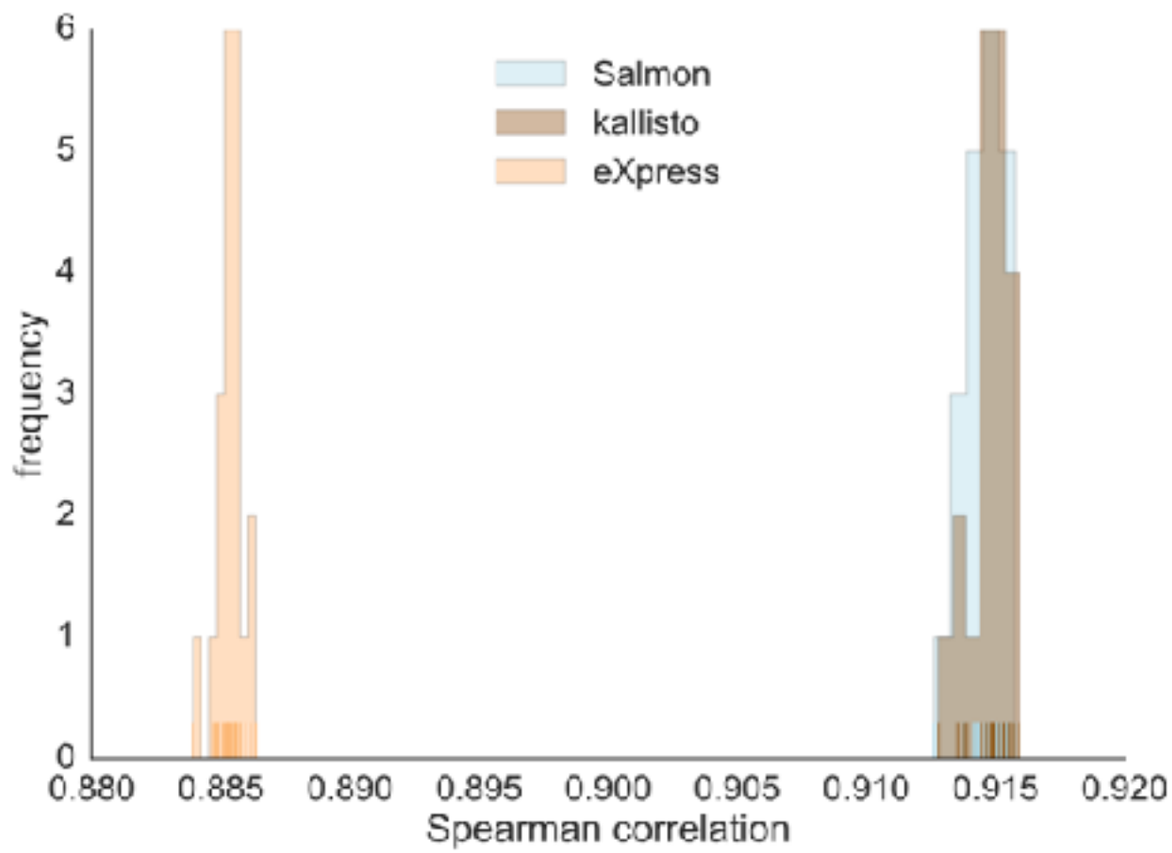
$$\hat{\eta}_i = \frac{\alpha_i}{\sum_j \alpha_j}$$

we also provide the option to use a variational Bayesian objective instead:

$$\alpha_i^{u+1} = \alpha_i^0 + \sum_{\boldsymbol{C}^j \in \boldsymbol{C}} d^j \left(\frac{e^{\gamma_i^u} w_i^j}{\sum_{t_k \in \boldsymbol{t}^j} e^{\gamma_k^u} w_k^j} \right) \quad \text{where} \quad \gamma_i^u = \Psi(\alpha_i^u) - \Psi\left(\sum_k \alpha_k^u \right)$$

Transcript inference methods can be very accurate





$$\mathrm{ARD}_i = \begin{cases} 0 & \text{if } x_i = y_i = 0 \\ \frac{|x_i - y_i|}{x_i + y_i} & \text{otherwise} \end{cases},$$

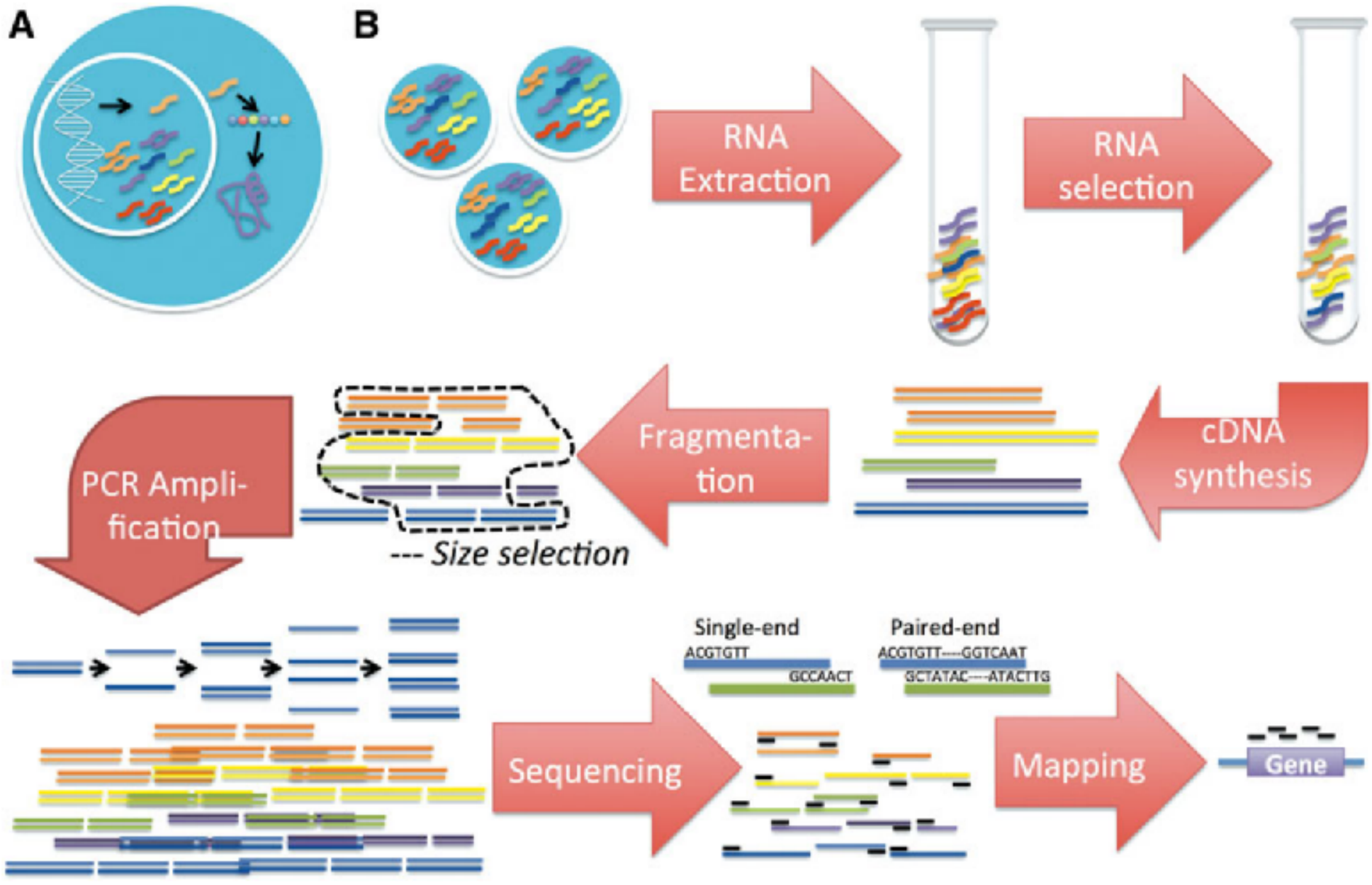
Results on 20 replicates simulated (RSEM-sim) from parameters learned from NA12716_7 from GEUVADIS. Showing result distributions for kallisto[1], eXpress[2] & salmon[3]

1: Bray, Nicolas L., et al. "Near-optimal probabilistic RNA-seq quantification." Nature biotechnology 34.5 (2016): 525-527. (v0.43.0)

2: Roberts, Adam, and Lior Pachter. "Streaming fragment assignment for real-time analysis of sequencing experiments." Nature methods 10.1 (2013): 71-73. (v.1.5.1)

3: Patro, Rob, et al. "Accurate, fast, and model-aware transcript expression quantification with Salmon." bioRxiv (2015): 021592. (v0.7.0)

Actual RNA-seq protocols are a bit more “involved”



There is substantial potential for biases and deviations from our model — indeed, we see quite a few.

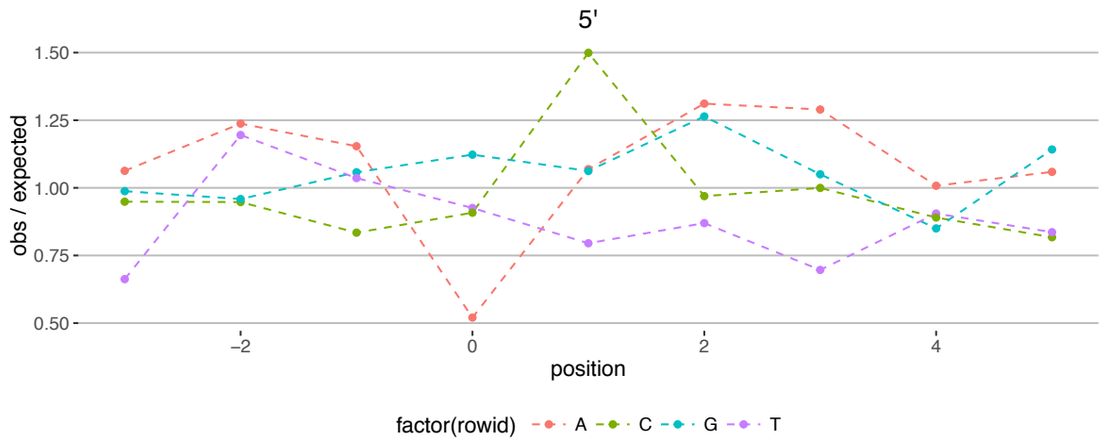
Biases abound in RNA-seq data

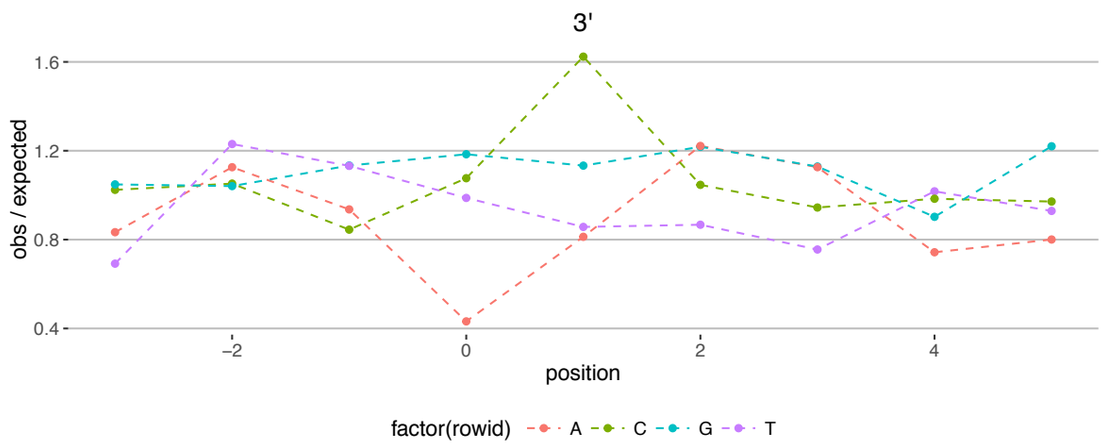
Biases in prep & sequencing can have a significant effect on the fragments we see.

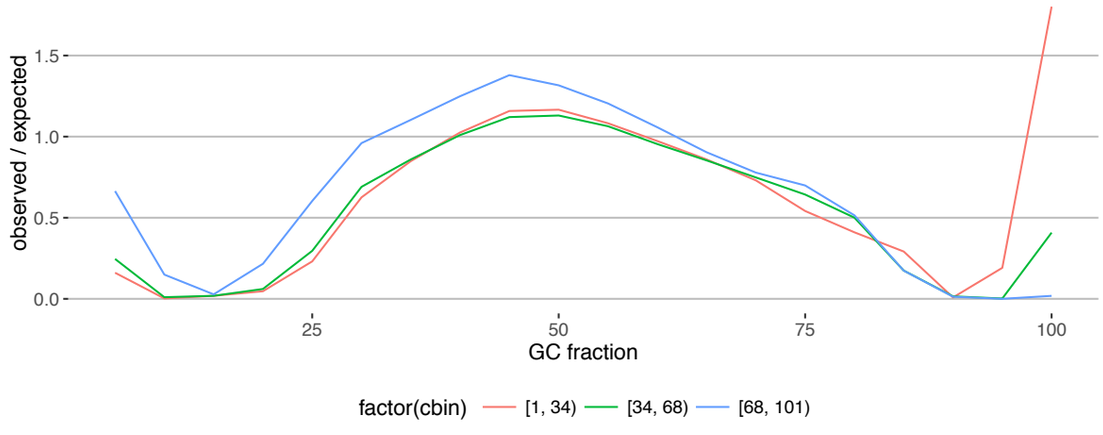
Fragment gc-bias[1]—
The GC-content of the fragment affects the likelihood of sequencing

Sequence-specific bias[2]—
sequences surrounding fragment affect the likelihood of sequencing

Positional bias[2]—
fragments sequenced non-uniformly across the body of a transcript







1:Love, Michael I., John B. Hogenesch, and Rafael A. Irizarry. "Modeling of RNA-seq fragment sequence bias reduces systematic errors in transcript abundance estimation." bioRxiv (2015): 025767.

2:Roberts, Adam, et al. "Improving RNA-Seq expression estimates by correcting for fragment bias." Genome biology 12.3 (2011): 1.

Bias Modeling

Bias correction works by adjusting the effective lengths of the transcripts:
The effective length becomes the sum of the per-base biases

$$\tilde{\ell}'_i = \sum_{j=1}^{j \leq \ell_i} \sum_{k=1}^{k \leq f_i(j,L)} \frac{b_{gc+}(t_i, j, j+k)}{b_{gc-}(t_i, j, j+k)} \cdot \frac{b^{5'}_{s+}(t_i, j)}{b^{5'}_{s-}(t_i, j)} \cdot \frac{b^{3'}_{s+}(t_i, j+k)}{b^{3'}_{s-}(t_i, j+k)} \cdot \frac{b^{5'}_{p+}(t_i, j+k)}{b^{5'}_{p-}(t_i, j+k)} \cdot \frac{b^{3'}_{p+}(t_i, j+k)}{b^{3'}_{p-}(t_i, j+k)} \cdot \Pr\{X = j\}$$

Fragment GC bias model:

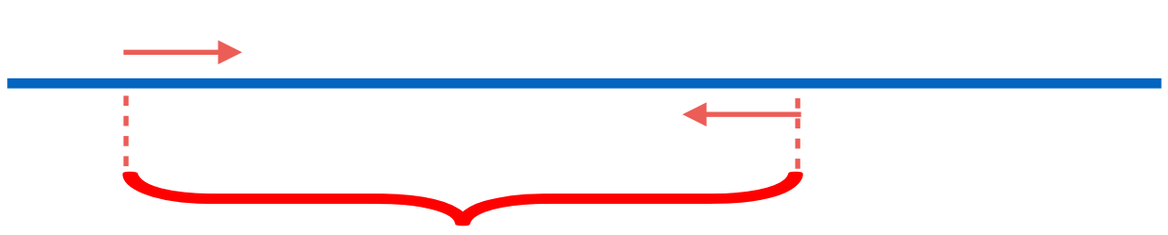
Density of fragments with specific GC content, **conditioned** on GC fraction at read start/end

Foreground:

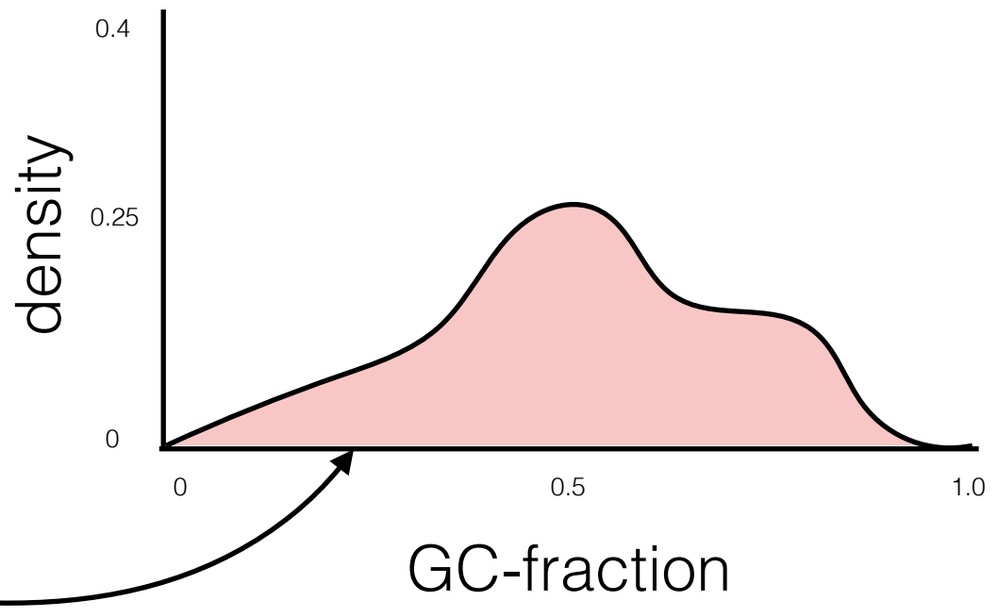
Observed

Background:

Expected given est. abundances



GC-fraction of fragment



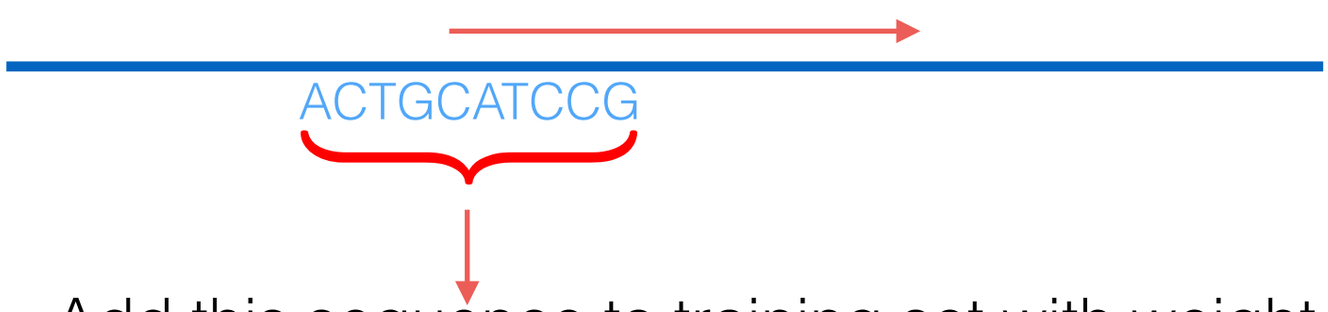
Bias Modeling

Bias correction works by adjusting the effective lengths of the transcripts: The effective length becomes the sum of the per-base biases

$$\tilde{\ell}'_i = \sum_{j=1}^{j \leq \ell_i} \sum_{k=1}^{k \leq f_i(j,L)} \frac{b_{gc+}(t_i, j, j+k)}{b_{gc-}(t_i, j, j+k)} \cdot \frac{b^{5'}_{s+}(t_i, j)}{b^{5'}_{s-}(t_i, j)} \cdot \frac{b^{3'}_{s+}(t_i, j+k)}{b^{3'}_{s-}(t_i, j+k)} \cdot \frac{b^{5'}_{p+}(t_i, j+k)}{b^{5'}_{p-}(t_i, j+k)} \cdot \frac{b^{3'}_{p+}(t_i, j+k)}{b^{3'}_{p-}(t_i, j+k)} \cdot \Pr\{X = j\}$$

Seq-specific bias model*:

VLMM for the 10bp window surrounding the 5’ read start site and the 3’ read start site



Add this sequence to training set with weight = $P\{f \mid t_i\}$

Foreground:

Observed

Background:

Expected given est. abundances

Same, but independent model for 3’ end

*Roberts, Adam, et al. "Improving RNA-Seq expression estimates by correcting for fragment bias." Genome biology 12.3 (2011): 1.

Bias Modeling

Bias correction works by adjusting the effective lengths of the transcripts: The effective length becomes the sum of the per-base biases

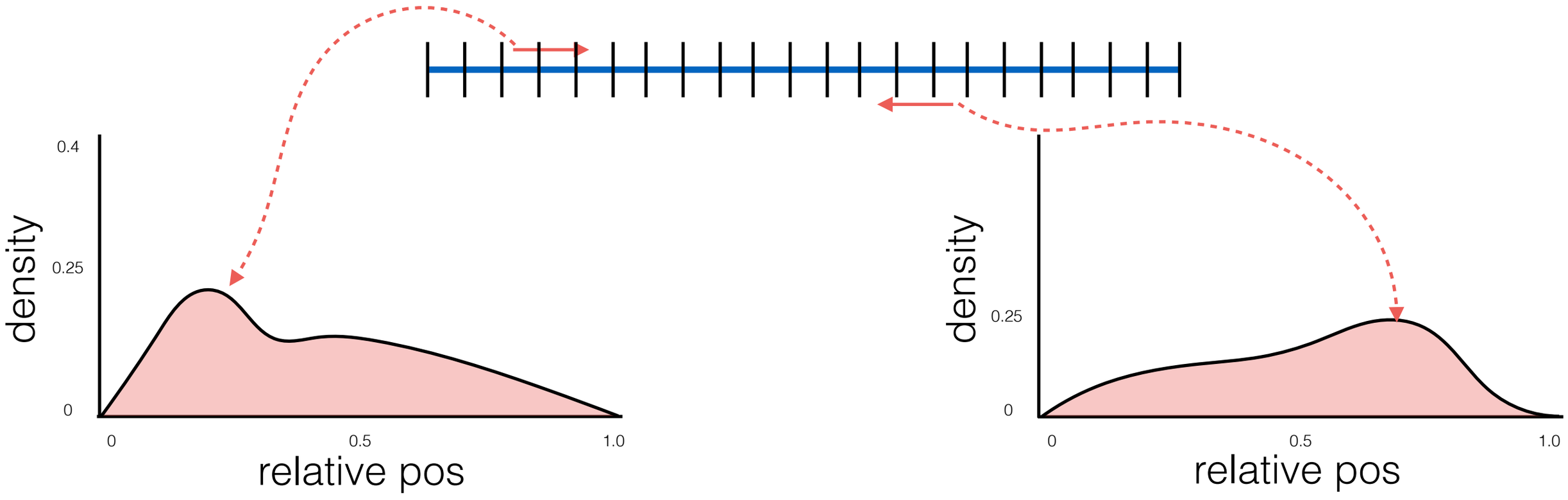
$$\tilde{\ell}'_i = \sum_{j=1}^{j \leq \ell_i} \sum_{k=1}^{k \leq f_i(j,L)} \frac{b_{gc^+}(t_i, j, j+k)}{b_{gc^-}(t_i, j, j+k)} \cdot \frac{b^{5'}_{s^+}(t_i, j)}{b^{5'}_{s^-}(t_i, j)} \cdot \frac{b^{3'}_{s^+}(t_i, j+k)}{b^{3'}_{s^-}(t_i, j+k)} \cdot \frac{b^{5'}_{p^+}(t_i, j+k)}{b^{5'}_{p^-}(t_i, j+k)} \cdot \frac{b^{3'}_{p^+}(t_i, j+k)}{b^{3'}_{p^-}(t_i, j+k)} \cdot \Pr\{X = j\}$$

Position bias model*:

Density of 5’ and 3’ read start positions — different models for transcripts of different length

Foreground:
Observed

Background:
Expected given est. abundances

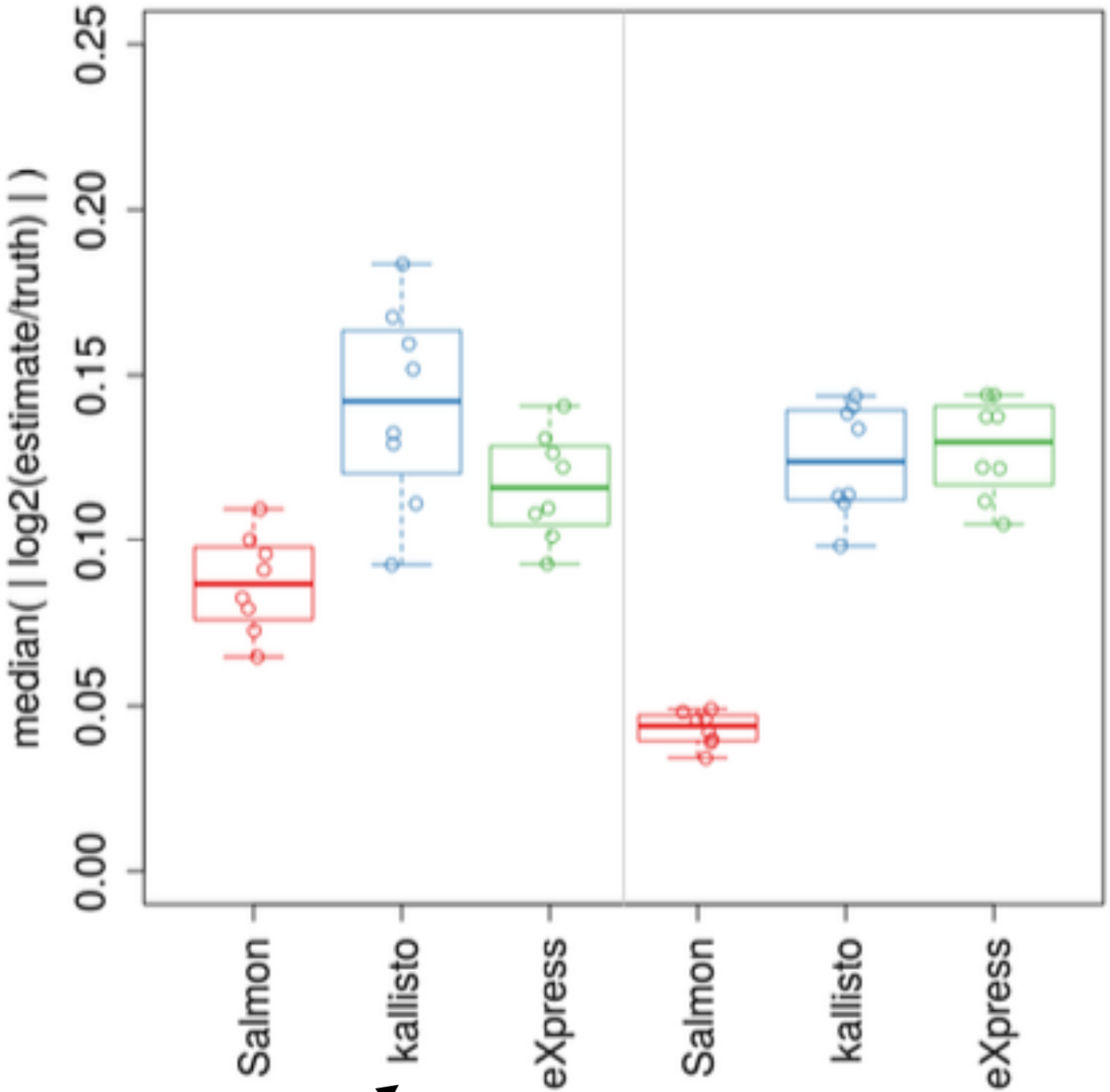


*Roberts, Adam, et al. "Improving RNA-Seq expression estimates by correcting for fragment bias." Genome biology 12.3 (2011): 1.

Accuracy difference can be larger with biased data

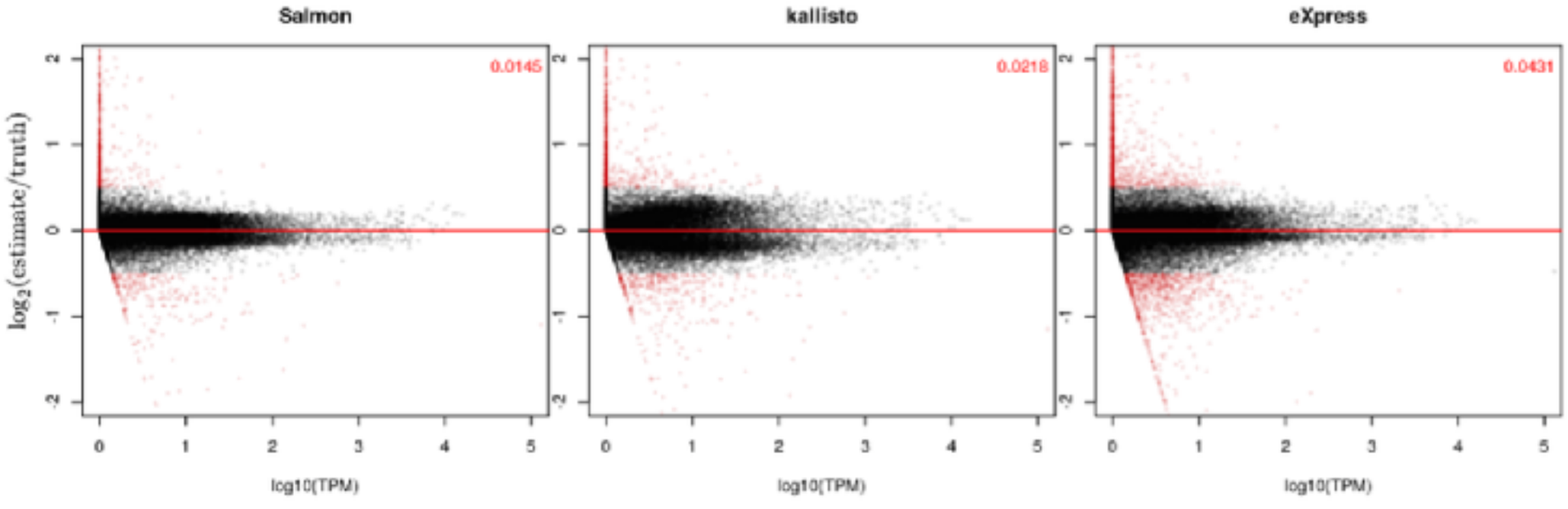
Simulated data:
2 conditions; 8 samples each

- Simulated transcripts across entire genome with known abundance using Polyester (modified to account for GC bias)
- How well do we recover the underlying relative abundances?
- How does accuracy vary with level of bias?



Sequence-bias models don’t account for fragment-level GC bias

Accuracy difference can be larger with biased data

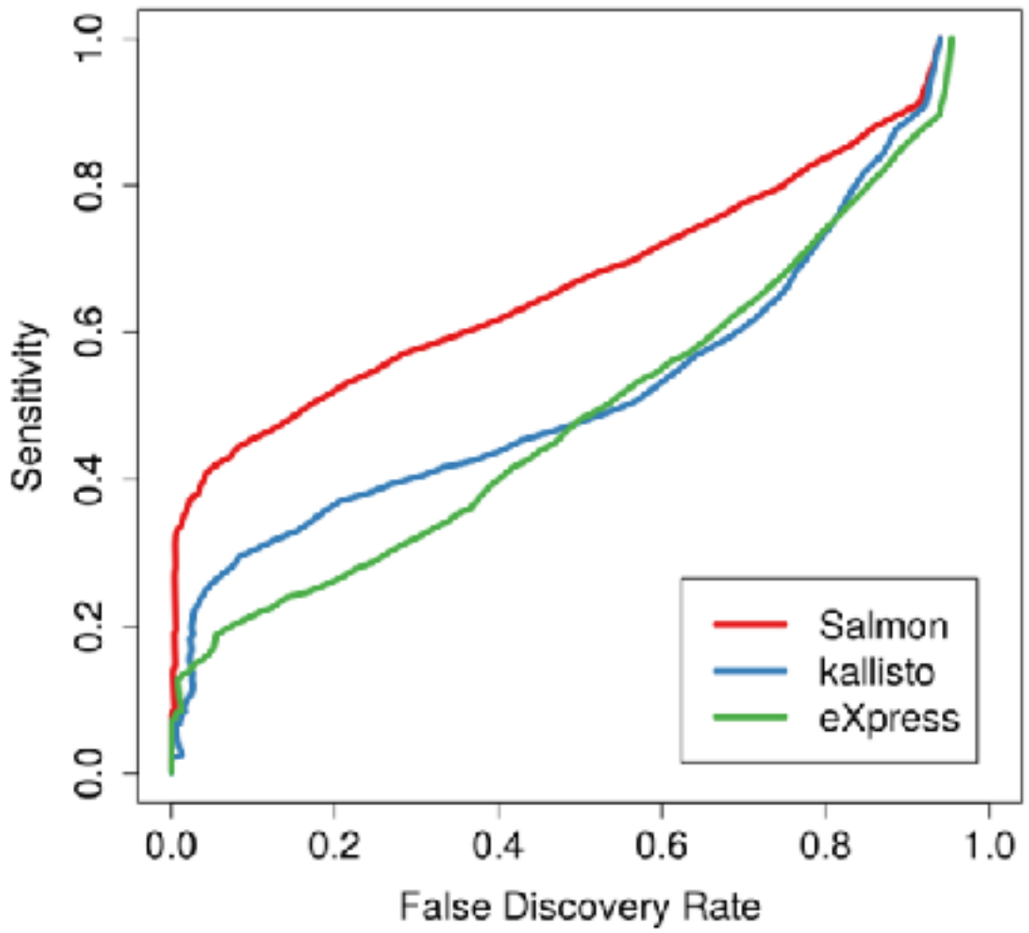


Mis-estimates confound downstream analysis

Simulated data:
2 conditions; 8 replicates each

- set 10% of txps to have fold change of 1/2 or 2 — rest unchanged.
- How well do we recover true DE?
- Since bias is systematic, effect may be even worse than accuracy difference suggests.

Recovery of DE transcripts

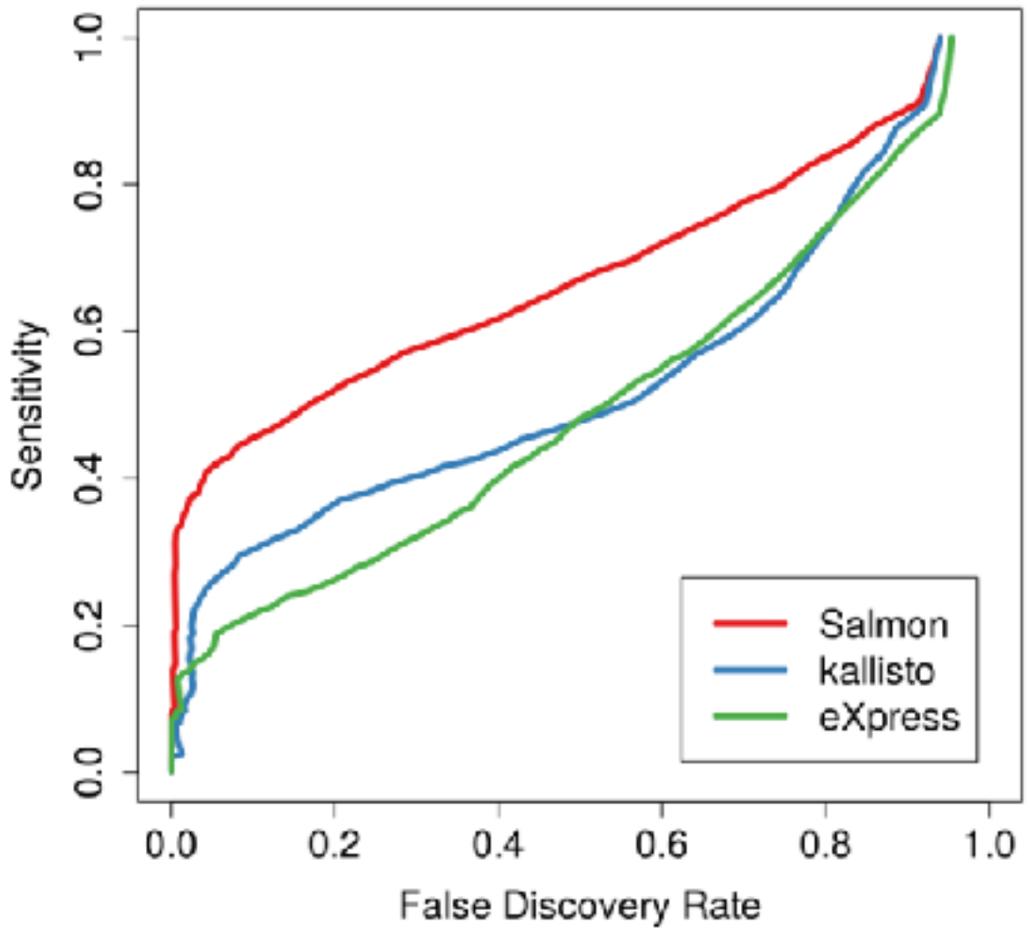


Accuracy difference can be large with biased data!

FDR	Sensitivity at given FDR		
	Salmon	*kallisto*	*eXpress*
0.01	0.326	0.072	0.128
0.05	0.409	0.248	0.162
0.1	0.454	0.296	0.211

At the same FDR, accuracy differences of 53 - 450%

Recovery of DE transcripts



Importance with experimental data

30 samples from the GEUVADIS study:
- 15 samples from UNIGE sequencing center
- 15 samples from CNAG_CRG sequencing center

Same human population, expect few-to-no *real* DE (primary differences in sample prep)

DE of data between centers (FDR < 1%) (TPM > 0.1)

	Salmon	Kallisto	eXpress
All transcripts	**1,171**	2,620	2,472
Transcripts of 2 isoform genes	**224**	545	531

Bias and **batch effects** are ***substantial***, and must be accounted for.

Importance with experimental data

30 samples from the GEUVADIS study:
  15 samples from UNIGE sequencing center
  15 samples from CNAG_CRG sequencing center

Same human population, expect few-to-no *real* DE (primary differences in sample prep)

DE of data between centers (FDR < 1%) (TPM > 0.1)

But this is txp-level DE, and I care only about **genes!**

	Salmon	Kallisto	eXpress
All transcripts	1,171	2,620	2,472
Transcripts of 2 isoform genes	224	545	531

Bias and **batch effects** are ***substantial***, and must be accounted for.

Importance with experimental data

30 samples from the GEUVADIS study:
15 samples from UNIGE sequencing center
15 samples from CNAG_CRG sequencing center

Effects seem **at least as extreme** at the gene level

DE of data between centers (FDR < 1%) (TPM > 0.1)

	Salmon	Kallisto	eXpress
All genes	**455**	1,200	1,582
Transcripts of 2 isoform genes	**228**	545	531

Bias and **batch effects** are ***substantial***, and must be accounted for.

Salmon and kallisto are FAST

http://www.sbnation.com/lookit/2016/8/12/12463026/katie-ledectky-800m-gold-video-highlights-rio-olympics

Salmon and kallisto are FAST

Consider the following test:

Take all 20 replicates of the RSEM-sim simulated data above, treat them as one, giant sample. This is 20 samples x 30M paired-end reads = 600 million read pairs or 1.2 billion individual reads.

Using 30 threads[1]:

kallisto can process this sample in 20 minutes
Salmon can process this sample in 23 minutes

Just *aligning* the reads to use e.g. eXpress, Cufflinks, RSEM etc. would take dozens of hours.

1: Intel Xeon E5-4600 (2.6GHz)

One “issue” with maximum likelihood (ML)

The generative statistical model is a principled and elegant way to represent the RNA-seq process.

It can be optimized efficiently using e.g. the EM / VBEM algorithm.

but, these efficient optimization algorithms return “point estimates” of the abundances. That is, there is no notion of how *certain* we are in the computed abundance of transcript.

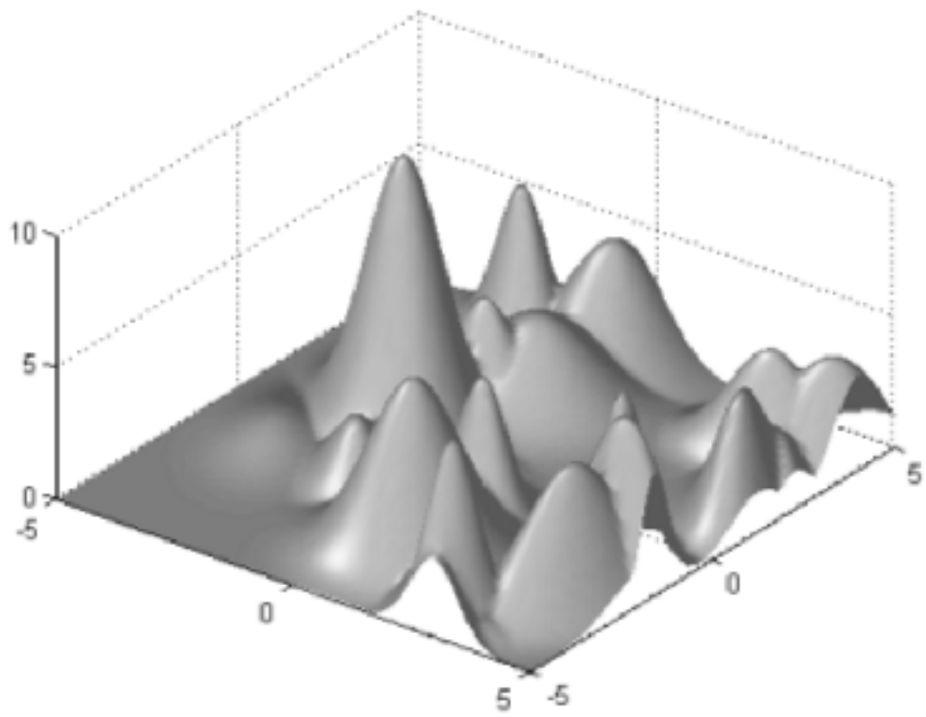
One “issue” with maximum likelihood (ML)

There are multiple sources of uncertainty e.g.

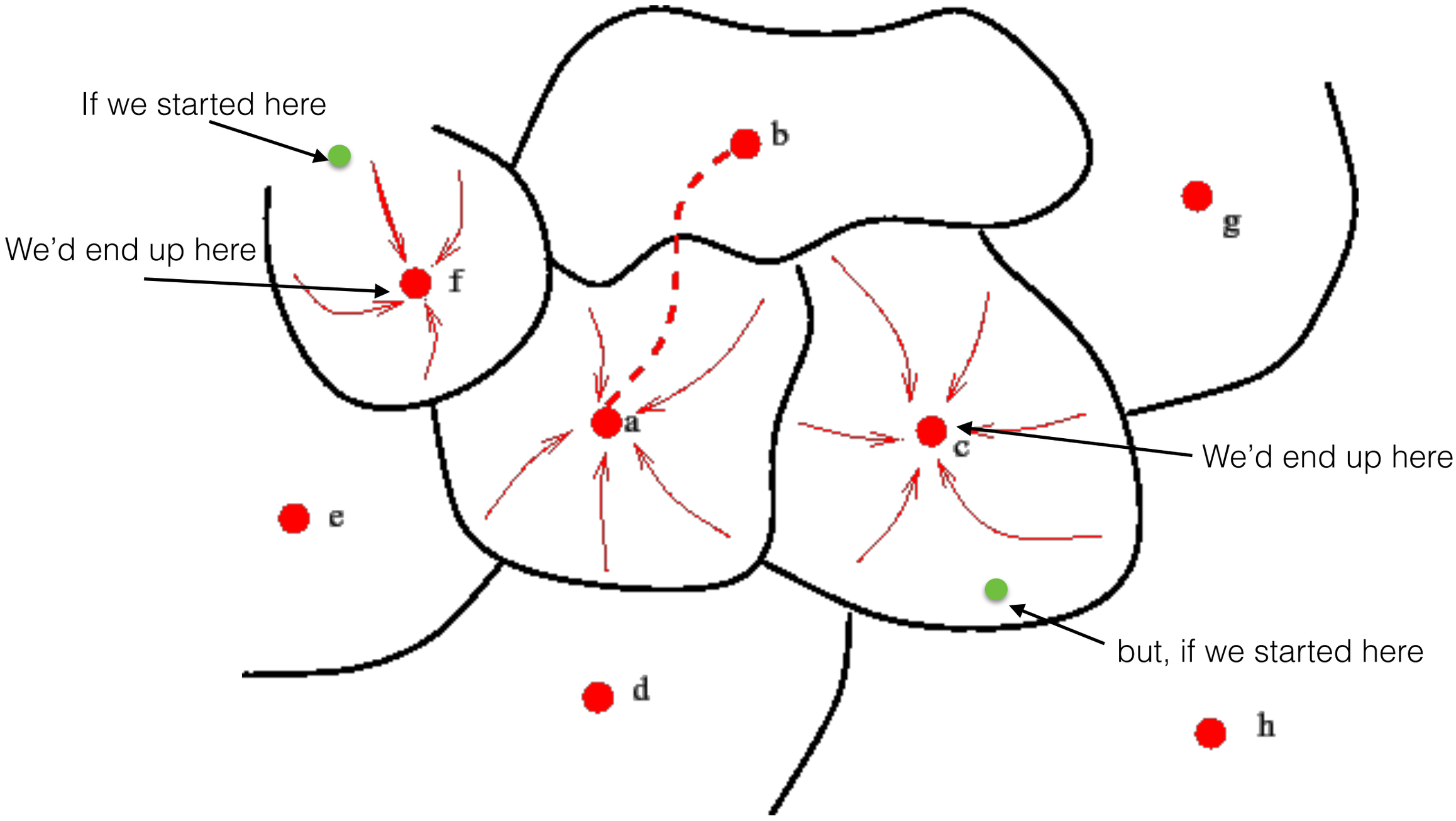
- Technical variance : If we sequenced the *exact* same sample again, we’d get a different set of fragments, and, potentially a different solution.

- Uncertainty in inference: We are almost never guaranteed to find a unique, globally optimal result. If we started our algorithm with different initialization parameters, we might get a different result.

We’re trying to find the *best* parameters in a space with 10s to 100s of thousands of dimensions!



One “issue” with maximum likelihood (ML)



https://commons.wikimedia.org/wiki/File:Local_search_attraction_basins.png (CC BY-SA 3.0)

Assessing Uncertainty

There are a few ways to address this “issue”

Do a fully Bayesian inference[1]:

Infer the entire posterior distribution of parameters, not just a ML estimate (e.g. using MCMC) — too slow!

✓ Posterior Gibbs Sampling[2,3]:

Starting from our ML estimate, do MCMC sampling to explore how parameters vary — if our ML estimate is good, and taking advantage of equivalence classes, this can be made *very fast.*

✓ Bootstrap Sampling[4]:

Resample (from equivalence class counts) with replacement, and re-run the ML estimate for each sample. This can be made reasonably fast.

Happy to discuss details / implications of this further.

1: BitSeq (with MCMC) actually does this. It’s very accurate, but very slow. [Glaus, Peter, Antti Honkela, and Magnus Rattray. "Identifying differentially expressed transcripts from RNA-seq data with biological variation." Bioinformatics 28.13 (2012): 1721-1728.]

2: RSEM has the ability to do this, and it seems to work well, but each sample scales in the # of reads. [Li, Bo, and Colin N. Dewey. "RSEM: accurate transcript quantification from RNA-Seq data with or without a reference genome." BMC bioinformatics 12.1 (2011): 1.]

3: MMSEQ can perform Gibbs sampling over shared variables (i.e. equiv classes), producing estimates from the mean of the posterior dist.Turro, Ernest, et al. "Haplotype and isoform specific expression estimation using multi-mapping RNA-seq reads." Genome biology 12.2 (2011): 1.

4: IsoDE introduced the idea of bootstrapping counts to assess quantification uncertainty. [Al Seesi, Sahar, et al. "Bootstrap-based differential gene expression analysis for RNA-Seq data with and without replicates." BMC genomics 15.8 (2014): 1.], but it was first made practical / fast in kallisto by doing the bootstrapping over equivalence classes.

Salmon addresses the main challenges of quantification

- finding locations of reads (mapping) is slow than necessary

→ Use quasi-mapping

- alternative splicing and related sequences creates ambiguity about where reads came from

→ Use dual-phase inference algorithm

- sampling of reads is not uniform or idealized

→ Use bias models learned from data

- uncertainty in ML estimate of abundances

→ Use posterior Gibbs sampling or bootstraps to assess uncertainty

Salmon has many other benefits

- Speed of inference makes it possible to use **bootstraps** or posterior **Gibbs sampling** to estimate variance (e.g. how certain are we in quantification estimates?).

- Quasi-mapping means no large, intermediate BAM files sitting on disk, or wasting computation time with slow disk I/O.

- Expressive model means new types of bias can be learned and accounted for.

- Separation of mapping / alignment and inference means Salmon can be used with or without existing alignments*. Here I talked only about quasi-mapping, but Salmon can use take BAM input from an aligner (if you really want!).

Many of these improvements (except dual-phase inference) have been back-ported to Sailfish, which is still *actively developed!*

https://github.com/kingsfordgroup/sailfish

Collaborators on Salmon

Geet Duggal (CMU / DNAnexus)

Carl Kingsford (CMU)

Mike Love (Harvard / UNC)

Rafael Irizarry(Harvard)

“Common” topics we didn’t get to cover:

RNA secondary structure prediction:

What structural features are present in an RNA-molecule?

Certain forms of the problem can be solved using DP.

Very interesting mixed DP & sampling solutions for harder variants.

Motif finding:

Finding “over represented” sub-sequences in a long sequence / collection of sequences.

Commonly approached via efficient statistical inference approaches (Gibbs sampling).

“Advanced” topics we didn’t get to cover:

Biological Networks

Inferring regulation from co-expression

Transferring knowledge by aligning networks

Network phylogeny — inferring ancestral networks?

Metagenomics

Estimating abundance of species from environmental sample

Assembling (partial) genomes from large environmental sample

How does metagenomic makeup change with phenotype?

Variant Detection and Association:

How do we efficiently and with high-confidence determine where a sequenced sample differs from a reference / different sample?

Which variations (groups of variations) are associated with certain traits, diseases or phenotypes?

“Advanced” topics we didn’t get to cover:

Too much else to list, but I’m happy to discuss with you!

Closing administrivia

Final: Dec 14th 5:30 — 8:00 PM (in this room)

Final project: Dec 15th 11:59 PM — please turn it in using Blackboard.
