Pt. 2 - Intro to Expression Quantification

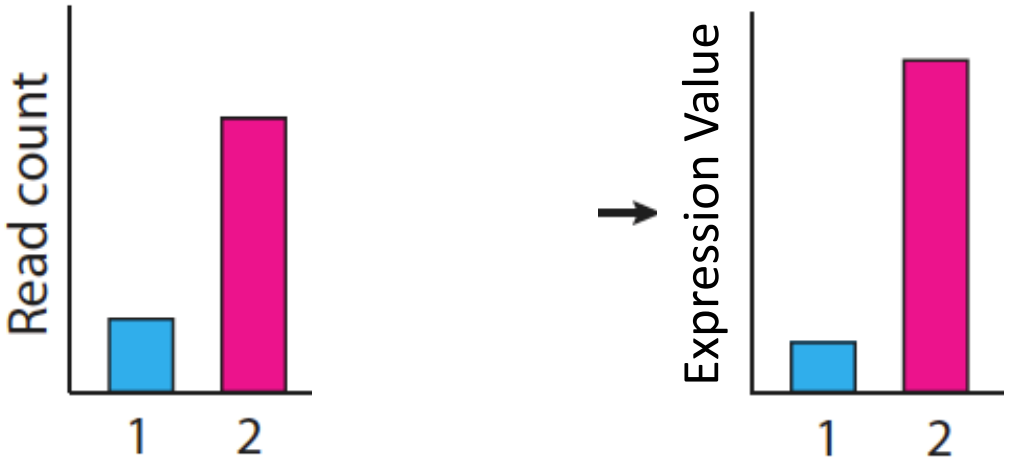
Programming for Biologists
CSHL 2019

Brian Haas

Broad Institute

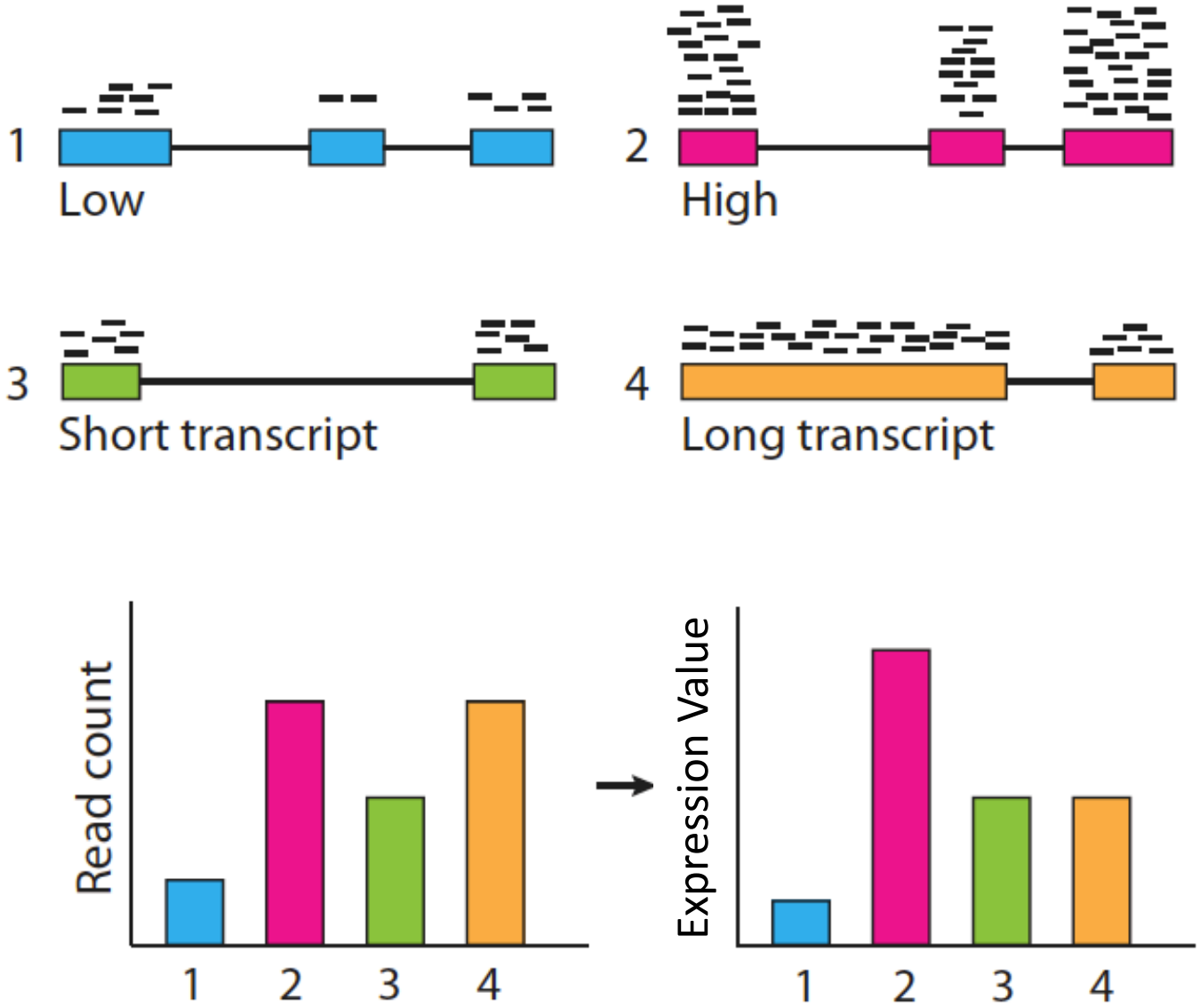
Image borrowed from (https://www.pharmaceutical-journal.com/pictures/600x424/9/6/5/1073965_man-measuring-mess-cr-ikonimages-17.png)

Calculating expression of genes and transcripts



Slide courtesy of Cole Trapnell

Calculating expression of genes and transcripts



Slide courtesy of Cole Trapnell

Normalized Expression Values

- Transcript-mapped read counts are normalized for both length of the transcript and total depth of sequencing.

- Reported as: Number of RNA-Seq **F**ragments
  Per **K**ilobase of transcript
  per total **M**illion fragments mapped

FPKM

RPKM (reads per kb per M) used with Single-end RNA-Seq reads
FPKM used with Paired-end RNA-Seq reads.

Transcripts per Million (TPM)

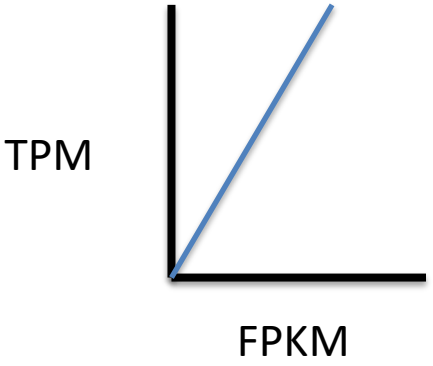
$$TPM_i = \frac{FPKM_i}{\sum_j FPKM} * 1e6$$

Preferred metric for measuring expression

- Better reflects transcript concentration in the sample.
- Nicely sums to 1 million

Linear relationship between TPM and FPKM values.

Both are valid metrics, but best to be consistent.



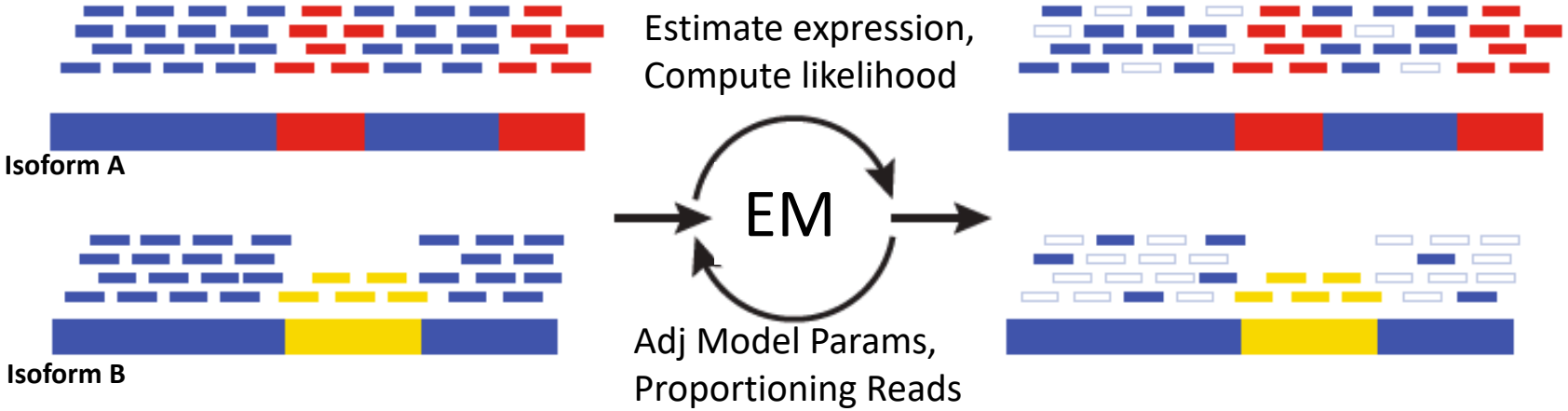
Multiply-mapped Reads Confound Abundance Estimation

Isoform A

Isoform B

Blue = multiply-mapped reads
Red, Yellow = uniquely-mapped reads

Multiply-mapped Reads Confound Abundance Estimation



Blue = multiply-mapped reads
Red, Yellow = uniquely-mapped reads

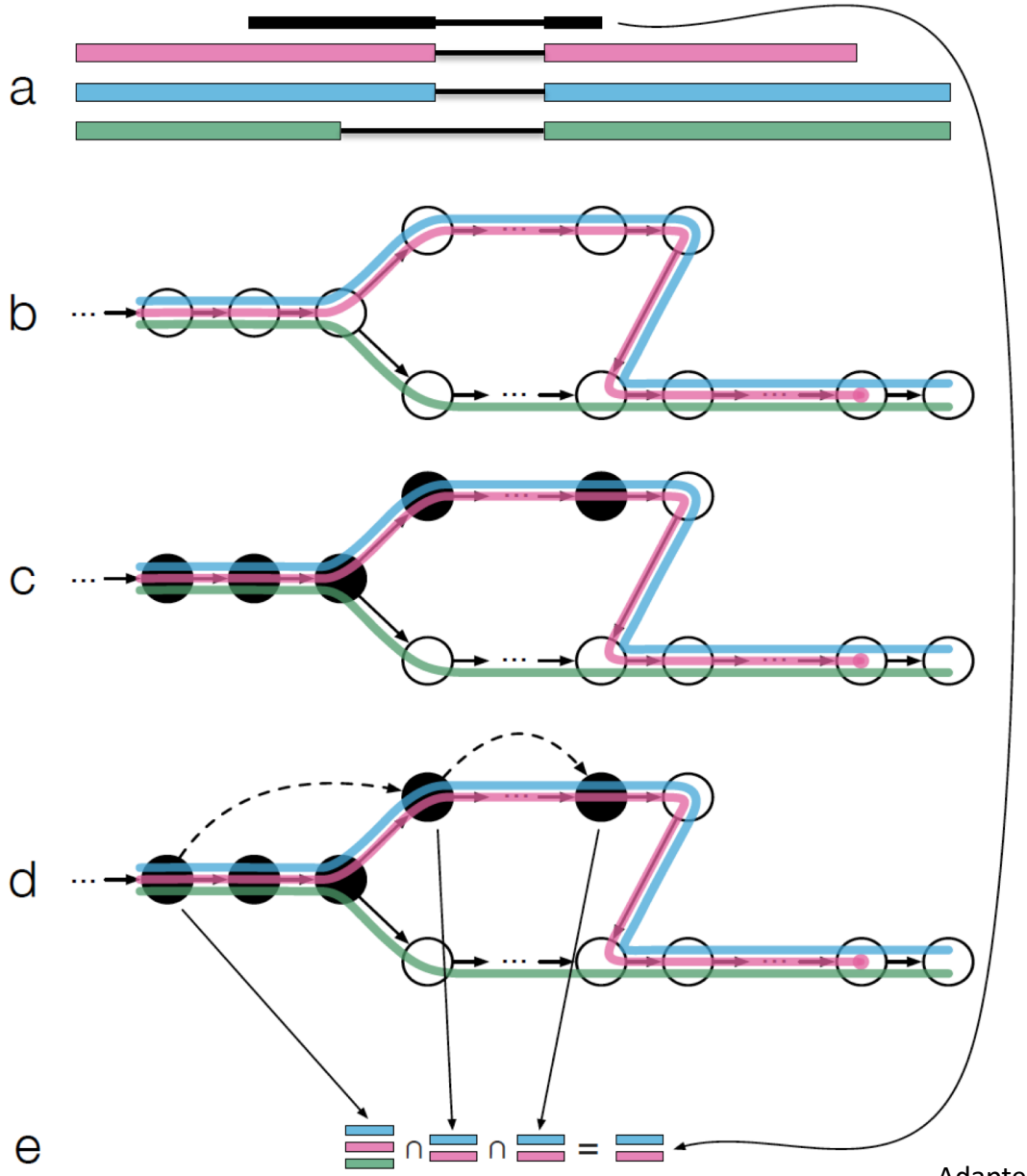
Use Expectation Maximization (EM) to find the most likely assignment of reads to transcripts.

Performed by:

- Cufflinks, String Tie (Tuxedo)
- RSEM, eXpress (genome-free)
- Kallisto, Salmon (alignment-free)

Fast Abundance Estimation Using Pseudo-alignments and Equivalence Classes

(Kallisto software, Bray et al., NBT 2016)



Adapted from Fig 1 from Bray et al.

ProgForBio Exercise 2: Build a utility that measures expression by counting aligned reads

Estimating Gene Expression Levels

Write a python program that reads in the 'bowtie2.bam' file and generates a table containing the number of reads mapped to each gene.

For example:

```
gene_read_counter.py bowtie2.bam
```

would return:

```
CG14995 1663
S-Lap3  1608
Eno     1423
sqd     877
AdipoR  801
Est-6   789
...
```

https://github.com/trinityrnaseq/CSHLProgForBiol2018/tree/master/Exercise_2-aligned_reads_to_expression

Summary of Key Points

- RNA-Seq is a versatile method for transcriptome analysis enabling quantification and novel transcript discovery.
- Expression quantification is based on sampling and counting reads derived from transcripts
- Trinity assembly and supported downstream computational analysis tools facilitate transcriptome studies.

RNA-Seq De novo Assembly Using Trinity

Visit website for more info:

http://trinityrnaseq.github.io

Quick Guide for the Impatient

Trinity assembles transcript sequences from Illumina RNA-Seq data.

Download Trinity here.

Build Trinity by typing 'make' in the base installation directory.

Assemble RNA-Seq data like so:

```
Trinity --seqType fq --left reads_1.fq --right reads_2.fq --CPU 6 --max_memory 20G
```

Find assembled transcripts as: 'trinity_out_dir/Trinity.fasta'

Use the documentation links in the right-sidebar to navigate this documentation, and contact our Google group for technical support.

Let’s go write some code! ☺