RNA Sequencing

Sam Kovaka
(Most slides by Michael Schatz)

Feb 27, 2019

Lecture 10: Applied Comparative Genomics

Review: Similarity metrics

- Hamming distance
  - Count the number of substitutions to transform one string into another

```
MIKESCHATZ
||x||xxxx|
MICESHATZZ
    5
```

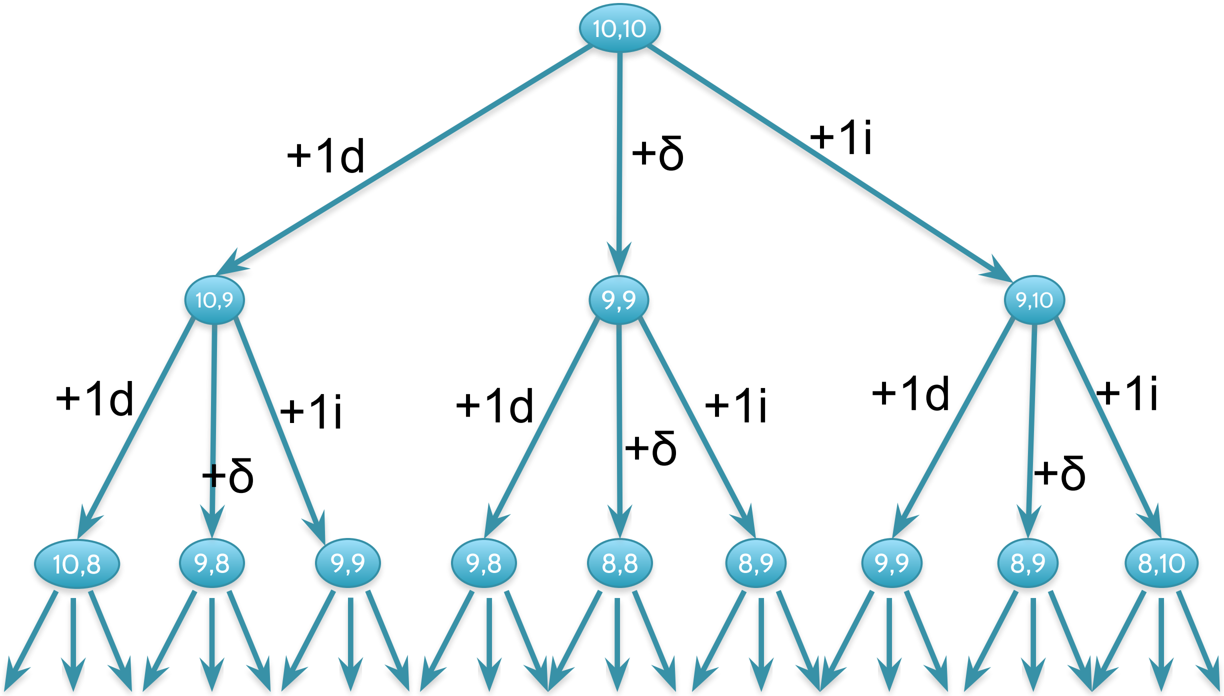
- Edit distance
  - The minimum number of substitutions, insertions, or deletions to transform one string into another

```
MIKESCHAT-Z
||x||x|||x|
MICES-HATZZ
     3
```

Recursive solution

- Computation of D is a recursive process.
  - At each step, we only allow matches, substitutions, and indels
  - D(i,j) in terms of D(i',j') for i' ≤ i and j' ≤ j.

D(MIKESCHATZ, MICESHATZZ) = min{D(MIKESCHATZ, MICESHATZ) + 1,
D(MIKESCHAT, MICESHATZZ) + 1,
D(MIKESCHAT, MICESHATZ) +δ(Z, Z)}



[What is the running time?]

Recurrence Relation for D

Find the edit distance (minimum number of sub, ins, del operations) to convert one string into another

- Base conditions:

  $D(i,0) = i$, for all $i = 0,\ldots,n$

  $D(0,j) = j$, for all $j = 0,\ldots,m$

- For $i > 0$, $j > 0$:

$$D(i,j) = \min \{$$

$D(i-1,j) + 1$, // align 0 from S, 1 from T

$D(i,j-1) + 1$, // align 1 from S, 0 from T

$D(i-1,j-1) + \delta(S(i),T(j))$ // align 1+1 chars

$$\}$$

[Why do we want the min?]

Dynamic Programming Matrix

		M	I	K	E	S	C	H	A	T	Z
	0	1	2	3	4	5	6	7	8	9	10
M	1										
I	2										
C	3										
E	4										
S	5										
H	6										
A	7										
T	8										
Z	9										
Z	10										

[What does the initialization mean?]

Dynamic Programming Matrix

		M	I	K	E	S	C	H	A	T	Z
	0	1	2	3	4	5	6	7	8	9	10
M	1	↖ ↑ ← 0									
I	2										
C	3										
E	4										
S	5										
H	6										
A	7										
T	8										
Z	9										
Z	10										

$$D[M,M] = \min\{D[M, \varnothing] +1,\ D[\varnothing,M]+1,\ D[\varnothing, \varnothing]+\delta(M,M)\}$$

Dynamic Programming Matrix

		M	I	K	E	S	C	H	A	T	Z
	0	1	2	3	4	5	6	7	8	9	10
M	1	0	1								
I	2										
C	3										
E	4										
S	5										
H	6										
A	7										
T	8										
Z	9										
Z	10										

$D[MI,M] = \min\{D[MI, \varnothing]+1, D[M,M]+1, D[M, \varnothing]+\delta(I,M)\}$

Dynamic Programming Matrix

		M	I	K	E	S	C	H	A	T	Z
	0	1	2	3	4	5	6	7	8	9	10
M	1	0	1	2							
I	2										
C	3										
E	4										
S	5										
H	6										
A	7										
T	8										
Z	9										
Z	10										

$D[MIK,M] = \min\{D[MIK, \varnothing]+1, D[MI,M]+1, D[MI,]+\delta(K,M)\}$

Dynamic Programming Matrix

		M	I	K	E	S	C	H	A	T	Z
	0	1	2	3	4	5	6	7	8	9	10
M	1	0	1	2	3						
I	2										
C	3										
E	4										
S	5										
H	6										
A	7										
T	8										
Z	9										
Z	10										

D[MIKE,M] = min{D[MIKE,]+1, D[MIK,M]+1, D[MIK,]+δ(E,M)}

Dynamic Programming Matrix

		M	I	K	E	S	C	H	A	T	Z
	0	1	2	3	4	5	6	7	8	9	10
M	1	0	1	2	3	4	5	6	7	8	9
I	2										
C	3										
E	4										
S	5										
H	6										
A	7										
T	8										
Z	9										
Z	10										

D[MIKESCHATZ,M] = 9

Dynamic Programming Matrix

		M	I	K	E	S	C	H	A	T	Z
	0	1	2	3	4	5	6	7	8	9	10
M	1	0	1	2	3	4	5	6	7	8	9
I	2	↖ ↑ ← 1									
C	3										
E	4										
S	5										
H	6										
A	7										
T	8										
Z	9										
Z	10										

$D[M,MI] = \min\{D[M,M]+1,\ D[MI, \varnothing]+1,\ D[\varnothing,M]+\delta(M,I)\}$

Dynamic Programming Matrix

		M	I	K	E	S	C	H	A	T	Z
	0	1	2	3	4	5	6	7	8	9	10
M	1	0	1	2	3	4	5	6	7	8	9
I	2	1	0								
C	3										
E	4										
S	5										
H	6										
A	7										
T	8										
Z	9										
Z	10										

$D[MI,MI] = \min\{D[MI,M]+1,\ D[M, MI]+1,\ D[M,M]+\delta(I,I)\}$

Dynamic Programming Matrix

		M	I	K	E	S	C	H	A	T	Z
	0	1	2	3	4	5	6	7	8	9	10
M	1	0	1	2	3	4	5	6	7	8	9
I	2	1	0	1							
C	3										
E	4										
S	5										
H	6										
A	7										
T	8										
Z	9										
Z	10										

D[MIK,MI] = min{D[MIK,M]+1, D[MI, MI]+1, D[MI,M]+δ(K,I)}

Dynamic Programming Matrix

		M	I	K	E	S	C	H	A	T	Z
	0	1	2	3	4	5	6	7	8	9	10
M	1	0	1	2	3	4	5	6	7	8	9
I	2	1	0	1	2	3	4	5	6	7	8
C	3										
E	4										
S	5										
H	6										
A	7										
T	8										
Z	9										
Z	10										

D[MIKESCHATZ,MI] = 8

Dynamic Programming Matrix

		M	I	K	E	S	C	H	A	T	Z
	0	1	2	3	4	5	6	7	8	9	10
M	1	0	1	2	3	4	5	6	7	8	9
I	2	1	0	1	2	3	4	5	6	7	8
C	3	2	1	1							
E	4										
S	5										
H	6										
A	7										
T	8										
Z	9										
Z	10										

D[MIK,MIC] = 1

Dynamic Programming Matrix

		M	I	K	E	S	C	H	A	T	Z
	0	1	2	3	4	5	6	7	8	9	10
M	1	0	1	2	3	4	5	6	7	8	9
I	2	1	0	1	2	3	4	5	6	7	8
C	3	2	1	1	2	3	3	4	5	6	7
E	4										
S	5										
H	6										
A	7										
T	8										
Z	9										
Z	10										

D[MIKESCHATZ,MIC] = 7

Dynamic Programming Matrix

		M	I	K	E	S	C	H	A	T	Z
	0	1	2	3	4	5	6	7	8	9	10
M	1	0	1	2	3	4	5	6	7	8	9
I	2	1	0	1	2	3	4	5	6	7	8
C	3	2	1	1	2	3	3	4	5	6	7
E	4	3	2	2	1	2	3	4	5	6	7
S	5										
H	6										
A	7										
T	8										
Z	9										
Z	10										

D[MIKESCHATZ,MICE] = 7

Dynamic Programming Matrix

		M	I	K	E	S	C	H	A	T	Z
	0	1	2	3	4	5	6	7	8	9	10
M	1	0	1	2	3	4	5	6	7	8	9
I	2	1	0	1	2	3	4	5	6	7	8
C	3	2	1	1	2	3	3	4	5	6	7
E	4	3	2	2	1	2	3	4	5	6	7
S	5	4	3	3	2	1	2	3	4	5	6
H	6	5	4	4	3	2	2	2	3	4	5
A	7	6	5	5	4	3	3	3	2	3	4
T	8	7	6	6	5	4	4	4	3	2	3
Z	9	8	7	7	6	5	5	5	4	3	2
Z	10	9	8	8	7	6	6	6	5	4	3

D[MIKESCHATZ,MICESHATZZ] = 3

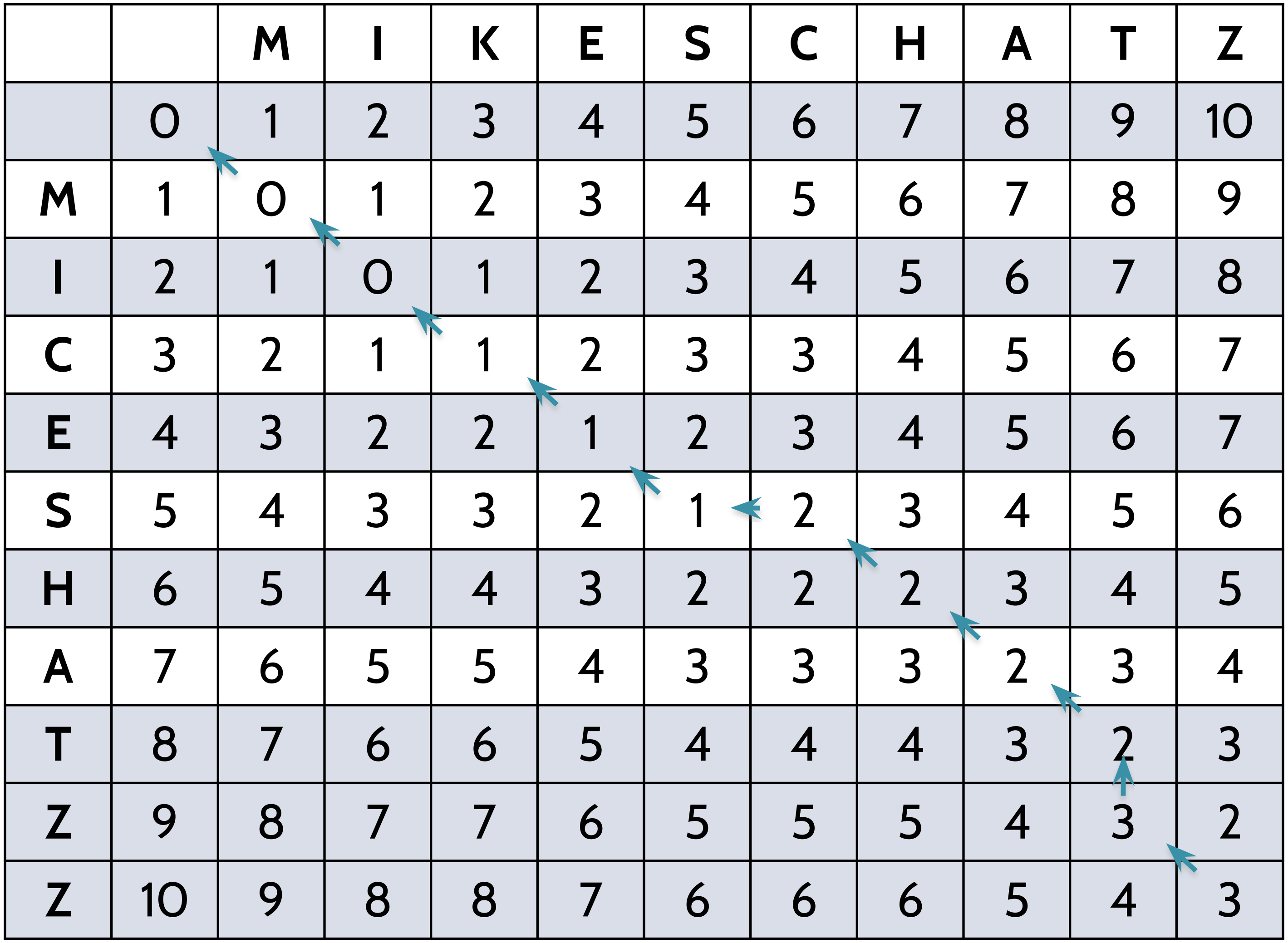
Dynamic Programming Matrix

		M	I	K	E	S	C	H	A	T	Z
	0	1	2	3	4	5	6	7	8	9	10
M	1	0	1	2	3	4	5	6	7	8	9
I	2	1	0	1	2	3	4	5	6	7	8
C	3	2	1	1	2	3	3	4	5	6	7
E	4	3	2	2	1	2	3	4	5	6	7
S	5	4	3	3	2	1	2	3	4	5	6
H	6	5	4	4	3	2	2	2	3	4	5
A	7	6	5	5	4	3	3	3	2	3	4
T	8	7	6	6	5	4	4	4	3	2	3
Z	9	8	7	7	6	5	5	5	4	3	2
Z	10	9	8	8	7	6	6	6	5	4	3

D[MIKESCHATZ,MICESHATZZ] = 3

Distance is 3, but how?

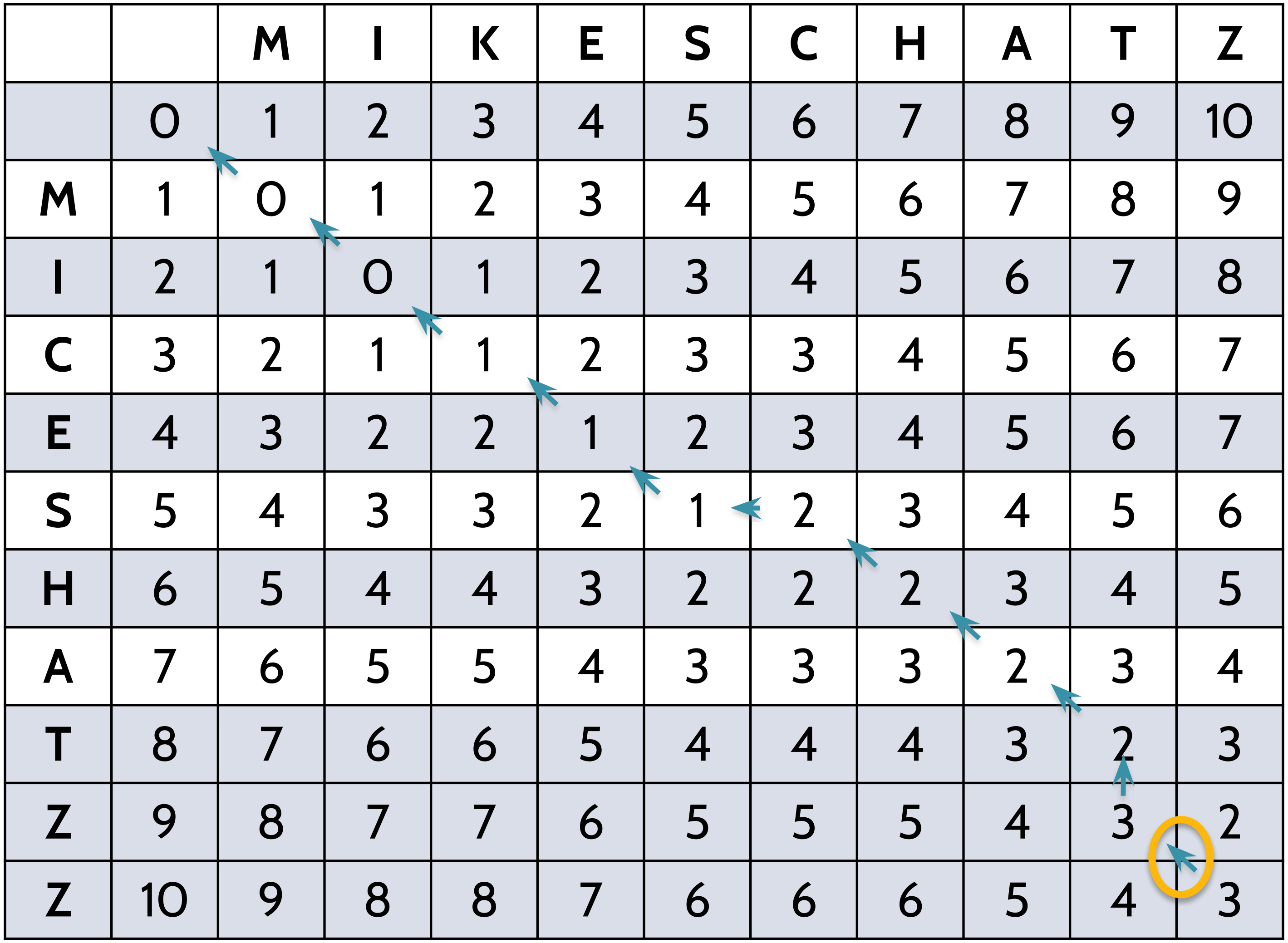
Dynamic Programming Matrix



		M	I	K	E	S	C	H	A	T	Z
	0	1	2	3	4	5	6	7	8	9	10
M	1	0	1	2	3	4	5	6	7	8	9
I	2	1	0	1	2	3	4	5	6	7	8
C	3	2	1	1	2	3	3	4	5	6	7
E	4	3	2	2	1	2	3	4	5	6	7
S	5	4	3	3	2	1	2	3	4	5	6
H	6	5	4	4	3	2	2	2	3	4	5
A	7	6	5	5	4	3	3	3	2	3	4
T	8	7	6	6	5	4	4	4	3	2	3
Z	9	8	7	7	6	5	5	5	4	3	2
Z	10	9	8	8	7	6	6	6	5	4	3

D[MIKESCHATZ,MICESHATZZ] = 3

Dynamic Programming Matrix



		M	I	K	E	S	C	H	A	T	Z
	0	1	2	3	4	5	6	7	8	9	10
M	1	0	1	2	3	4	5	6	7	8	9
I	2	1	0	1	2	3	4	5	6	7	8
C	3	2	1	1	2	3	3	4	5	6	7
E	4	3	2	2	1	2	3	4	5	6	7
S	5	4	3	3	2	1	2	3	4	5	6
H	6	5	4	4	3	2	2	2	3	4	5
A	7	6	5	5	4	3	3	3	2	3	4
T	8	7	6	6	5	4	4	4	3	2	3
Z	9	8	7	7	6	5	5	5	4	3	2
Z	10	9	8	8	7	6	6	6	5	4	3

Line up chars

D[MIKESCHATZ,MICESHATZZ] = 3

Z
Z

Dynamic Programming Matrix

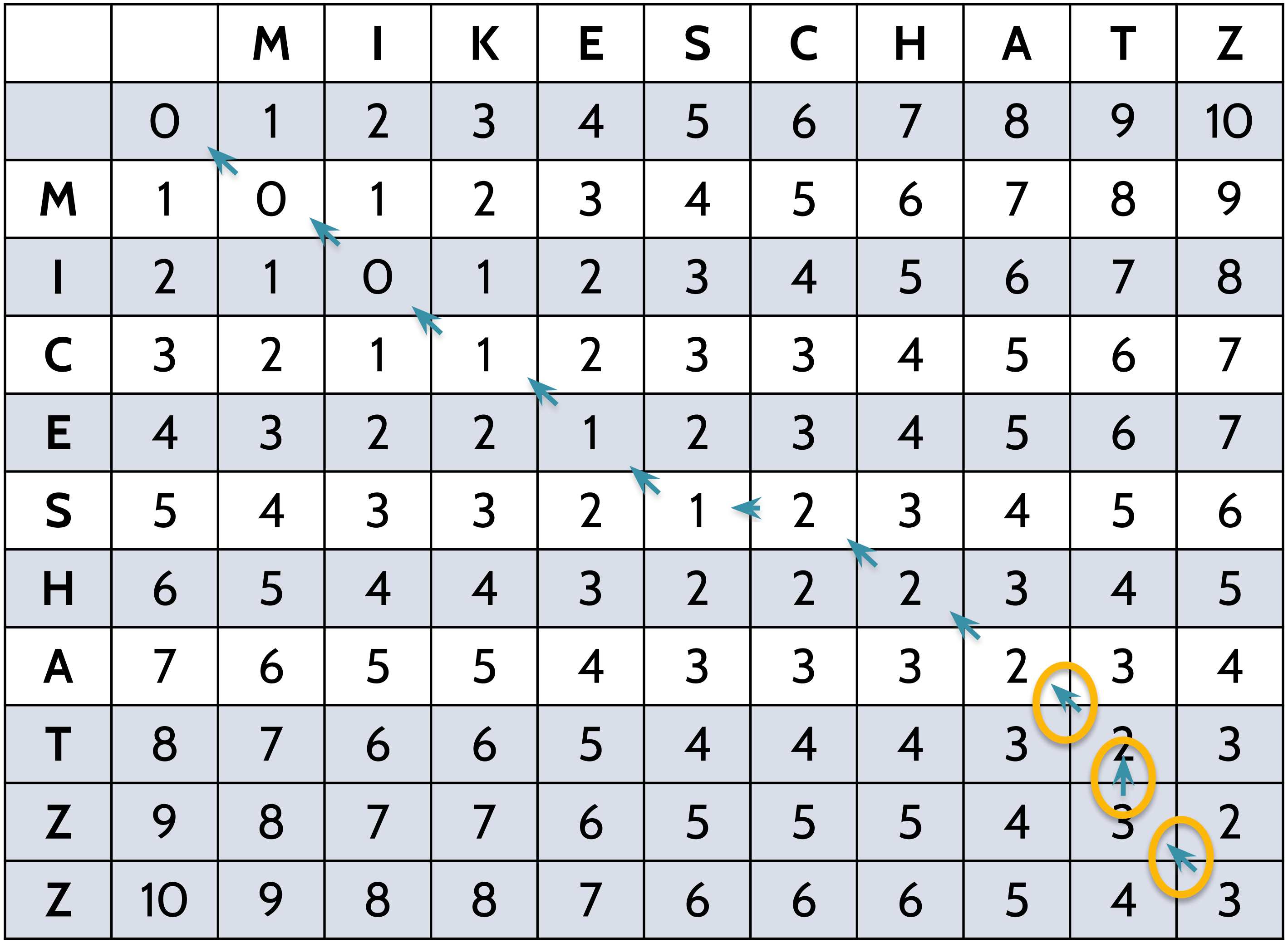
		M	I	K	E	S	C	H	A	T	Z
	0	1	2	3	4	5	6	7	8	9	10
M	1	0	1	2	3	4	5	6	7	8	9
I	2	1	0	1	2	3	4	5	6	7	8
C	3	2	1	1	2	3	3	4	5	6	7
E	4	3	2	2	1	2	3	4	5	6	7
S	5	4	3	3	2	1	2	3	4	5	6
H	6	5	4	4	3	2	2	2	3	4	5
A	7	6	5	5	4	3	3	3	2	3	4
T	8	7	6	6	5	4	4	4	3	2	3
Z	9	8	7	7	6	5	5	5	4	3	2
Z	10	9	8	8	7	6	6	6	5	4	3

Gap in top string

D[MIKESCHATZ,MICESHATZZ] = 3

-Z
ZZ

Dynamic Programming Matrix



		M	I	K	E	S	C	H	A	T	Z
	0	1	2	3	4	5	6	7	8	9	10
M	1	0	1	2	3	4	5	6	7	8	9
I	2	1	0	1	2	3	4	5	6	7	8
C	3	2	1	1	2	3	3	4	5	6	7
E	4	3	2	2	1	2	3	4	5	6	7
S	5	4	3	3	2	1	2	3	4	5	6
H	6	5	4	4	3	2	2	2	3	4	5
A	7	6	5	5	4	3	3	3	2	3	4
T	8	7	6	6	5	4	4	4	3	2	3
Z	9	8	7	7	6	5	5	5	4	3	2
Z	10	9	8	8	7	6	6	6	5	4	3

D[MIKESCHATZ,MICESHATZZ] = 3

```
T-Z
TZZ
```
]

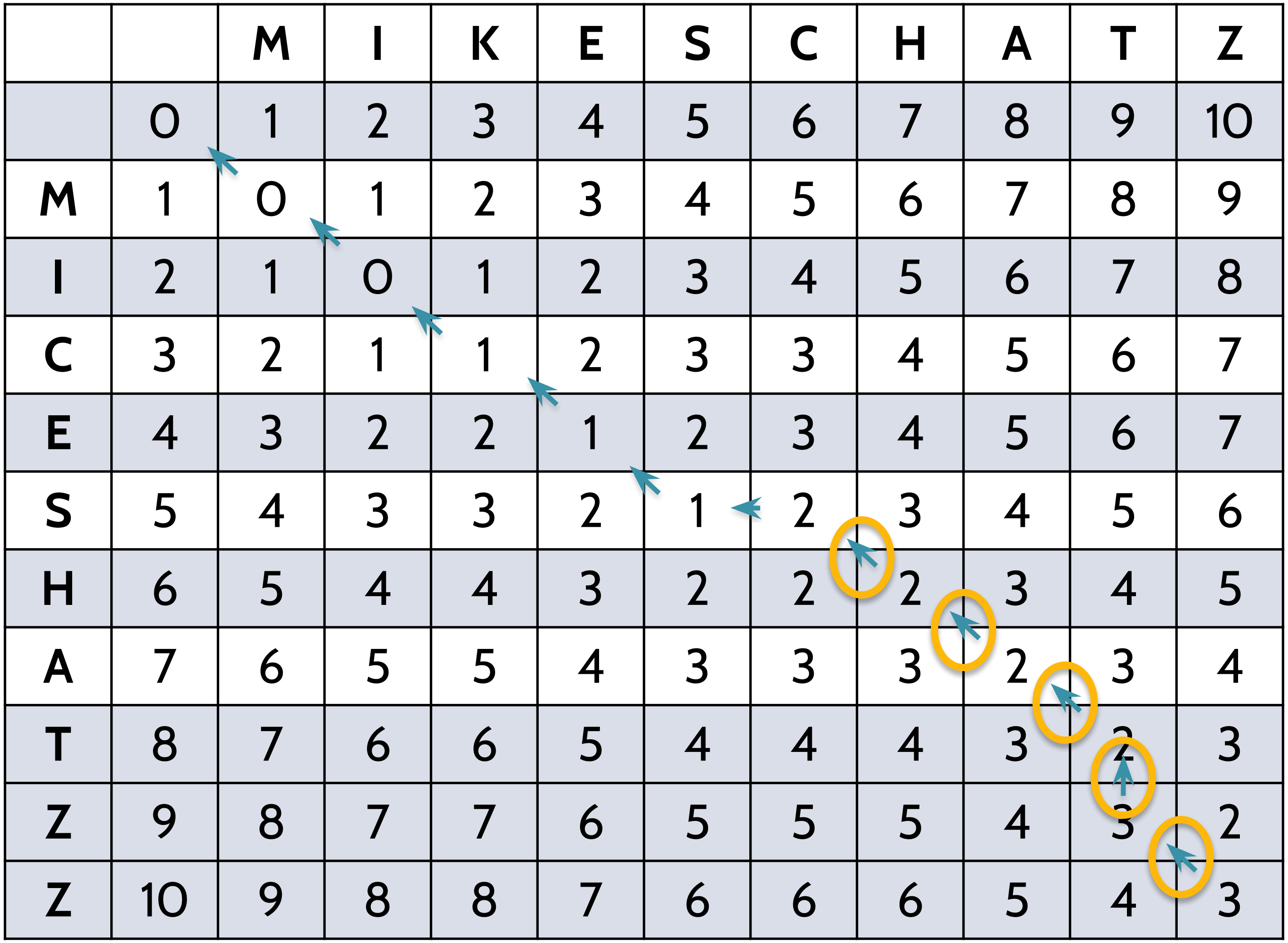
Dynamic Programming Matrix

		M	I	K	E	S	C	H	A	T	Z
	0	1	2	3	4	5	6	7	8	9	10
M	1	0	1	2	3	4	5	6	7	8	9
I	2	1	0	1	2	3	4	5	6	7	8
C	3	2	1	1	2	3	3	4	5	6	7
E	4	3	2	2	1	2	3	4	5	6	7
S	5	4	3	3	2	1	2	3	4	5	6
H	6	5	4	4	3	2	2	2	3	4	5
A	7	6	5	5	4	3	3	3	2	3	4
T	8	7	6	6	5	4	4	4	3	2	3
Z	9	8	7	7	6	5	5	5	4	3	2
Z	10	9	8	8	7	6	6	6	5	4	3

D[MIKESCHATZ,MICESHATZZ] = 3

```
AT-Z
ATZZ
```

Dynamic Programming Matrix

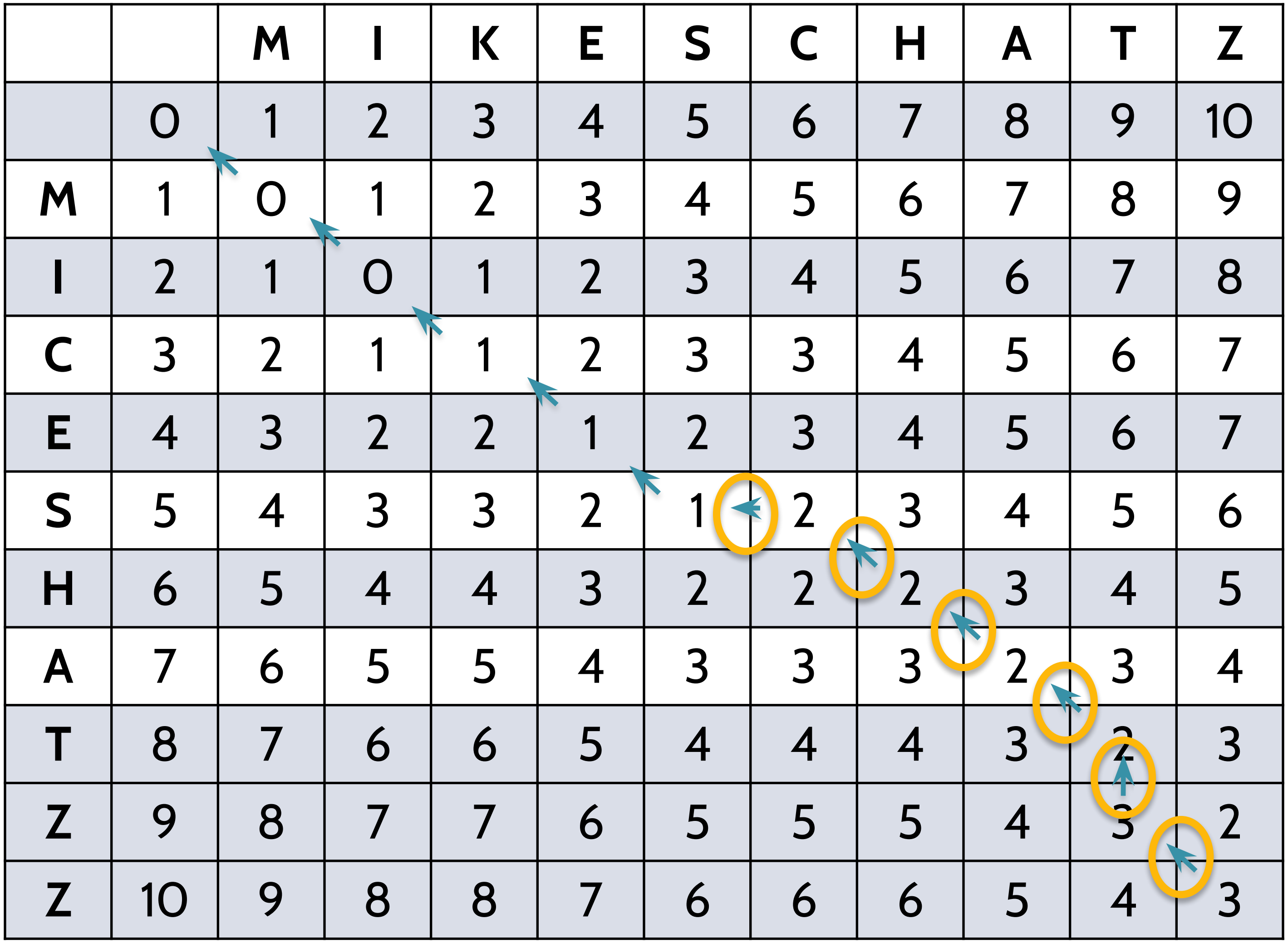


		M	I	K	E	S	C	H	A	T	Z
	0	1	2	3	4	5	6	7	8	9	10
M	1	0	1	2	3	4	5	6	7	8	9
I	2	1	0	1	2	3	4	5	6	7	8
C	3	2	1	1	2	3	3	4	5	6	7
E	4	3	2	2	1	2	3	4	5	6	7
S	5	4	3	3	2	1	2	3	4	5	6
H	6	5	4	4	3	2	2	2	3	4	5
A	7	6	5	5	4	3	3	3	2	3	4
T	8	7	6	6	5	4	4	4	3	2	3
Z	9	8	7	7	6	5	5	5	4	3	2
Z	10	9	8	8	7	6	6	6	5	4	3

D[MIKESCHATZ,MICESHATZZ] = 3

```
HAT-Z
HATZZ
```

Dynamic Programming Matrix



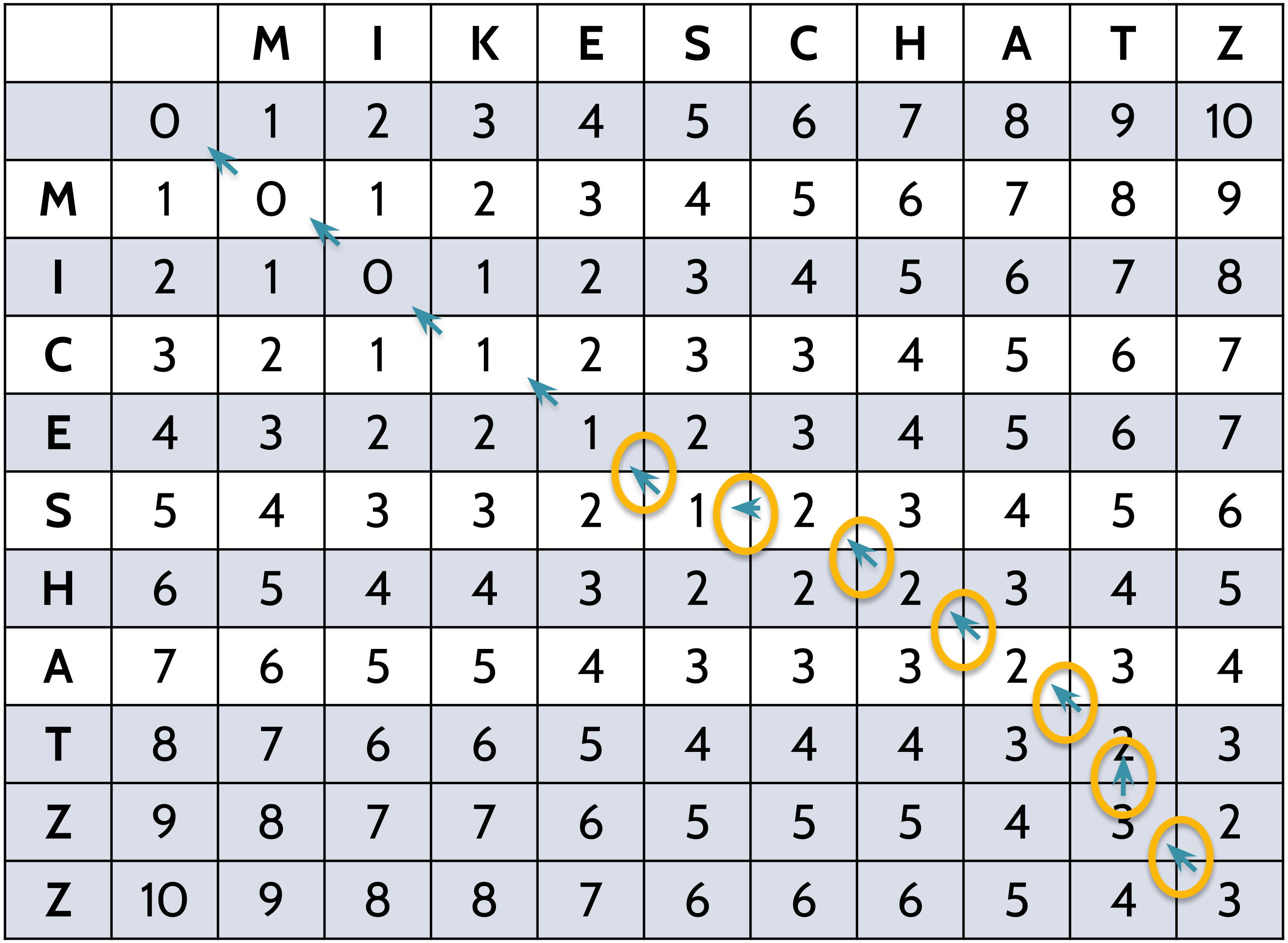
		M	I	K	E	S	C	H	A	T	Z
	0	1	2	3	4	5	6	7	8	9	10
M	1	0	1	2	3	4	5	6	7	8	9
I	2	1	0	1	2	3	4	5	6	7	8
C	3	2	1	1	2	3	3	4	5	6	7
E	4	3	2	2	1	2	3	4	5	6	7
S	5	4	3	3	2	1	2	3	4	5	6
H	6	5	4	4	3	2	2	2	3	4	5
A	7	6	5	5	4	3	3	3	2	3	4
T	8	7	6	6	5	4	4	4	3	2	3
Z	9	8	7	7	6	5	5	5	4	3	2
Z	10	9	8	8	7	6	6	6	5	4	3

Gap in bottom string

D[MIKESCHATZ,MICESHATZZ] = 3

```
CHAT-Z
-HATZZ
```

Dynamic Programming Matrix



		M	I	K	E	S	C	H	A	T	Z
	0	1	2	3	4	5	6	7	8	9	10
M	1	0	1	2	3	4	5	6	7	8	9
I	2	1	0	1	2	3	4	5	6	7	8
C	3	2	1	1	2	3	3	4	5	6	7
E	4	3	2	2	1	2	3	4	5	6	7
S	5	4	3	3	2	1	2	3	4	5	6
H	6	5	4	4	3	2	2	2	3	4	5
A	7	6	5	5	4	3	3	3	2	3	4
T	8	7	6	6	5	4	4	4	3	2	3
Z	9	8	7	7	6	5	5	5	4	3	2
Z	10	9	8	8	7	6	6	6	5	4	3

D[MIKESCHATZ,MICESHATZZ] = 3

```
SCHAT-Z
S-HATZZ
```

Dynamic Programming Matrix

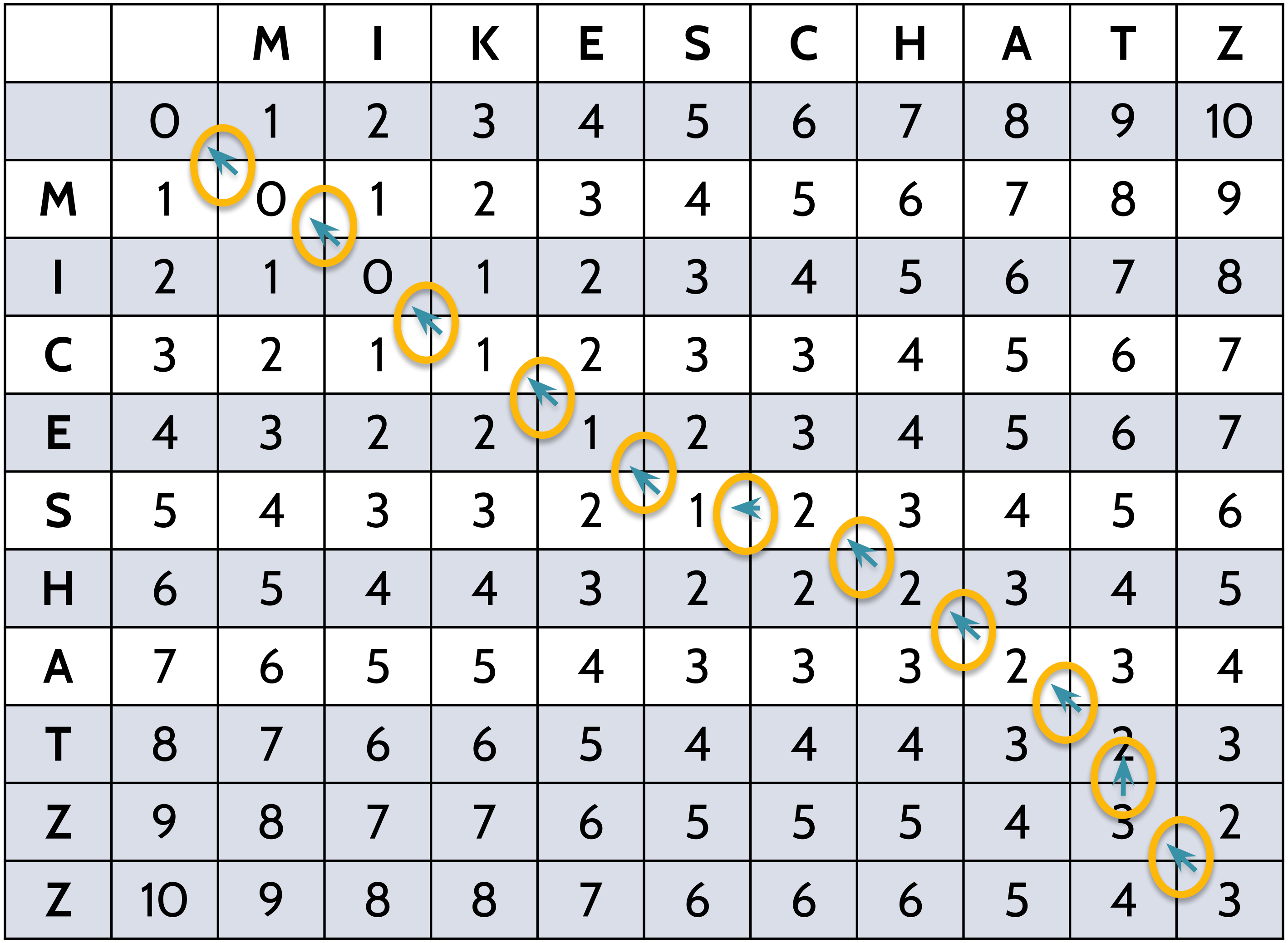
		M	I	K	E	S	C	H	A	T	Z
	0	1	2	3	4	5	6	7	8	9	10
M	1	0	1	2	3	4	5	6	7	8	9
I	2	1	0	1	2	3	4	5	6	7	8
C	3	2	1	1	2	3	3	4	5	6	7
E	4	3	2	2	1	2	3	4	5	6	7
S	5	4	3	3	2	1	2	3	4	5	6
H	6	5	4	4	3	2	2	2	3	4	5
A	7	6	5	5	4	3	3	3	2	3	4
T	8	7	6	6	5	4	4	4	3	2	3
Z	9	8	7	7	6	5	5	5	4	3	2
Z	10	9	8	8	7	6	6	6	5	4	3

Just line up mis-matches

D[MIKESCHATZ,MICESHATZZ] = 3

```
KESCHAT-Z
CES-HATZZ
```

Dynamic Programming Matrix



		M	I	K	E	S	C	H	A	T	Z
	0	1	2	3	4	5	6	7	8	9	10
M	1	0	1	2	3	4	5	6	7	8	9
I	2	1	0	1	2	3	4	5	6	7	8
C	3	2	1	1	2	3	3	4	5	6	7
E	4	3	2	2	1	2	3	4	5	6	7
S	5	4	3	3	2	1	2	3	4	5	6
H	6	5	4	4	3	2	2	2	3	4	5
A	7	6	5	5	4	3	3	3	2	3	4
T	8	7	6	6	5	4	4	4	3	2	3
Z	9	8	7	7	6	5	5	5	4	3	2
Z	10	9	8	8	7	6	6	6	5	4	3

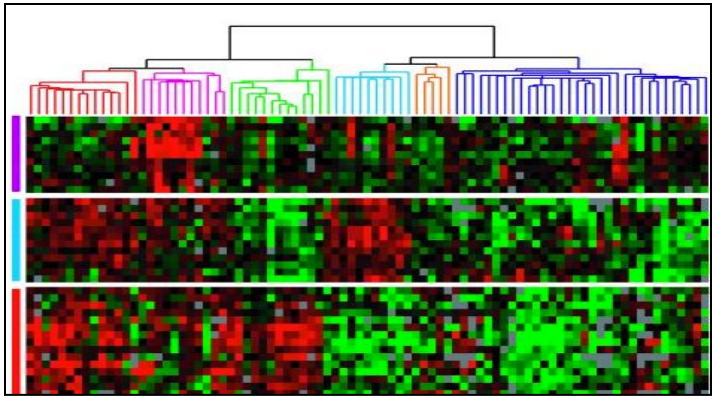
D[MIKESCHATZ,MICESHATZZ] = 3

```
MIKESCHAT-Z
MICES-HATZZ
```

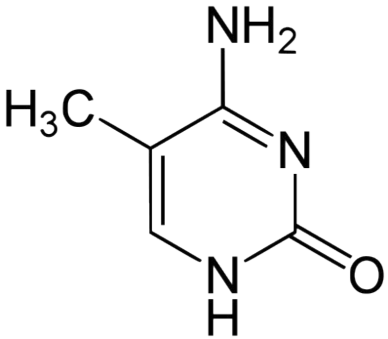
Hooray!

*-seq in 4 short vignettes

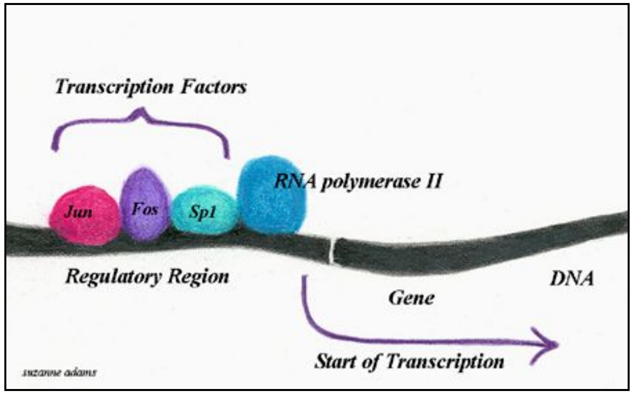
RNA-seq



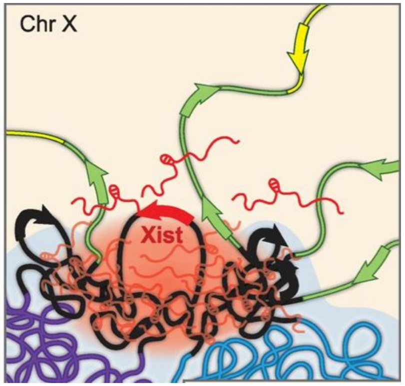
Methyl-seq



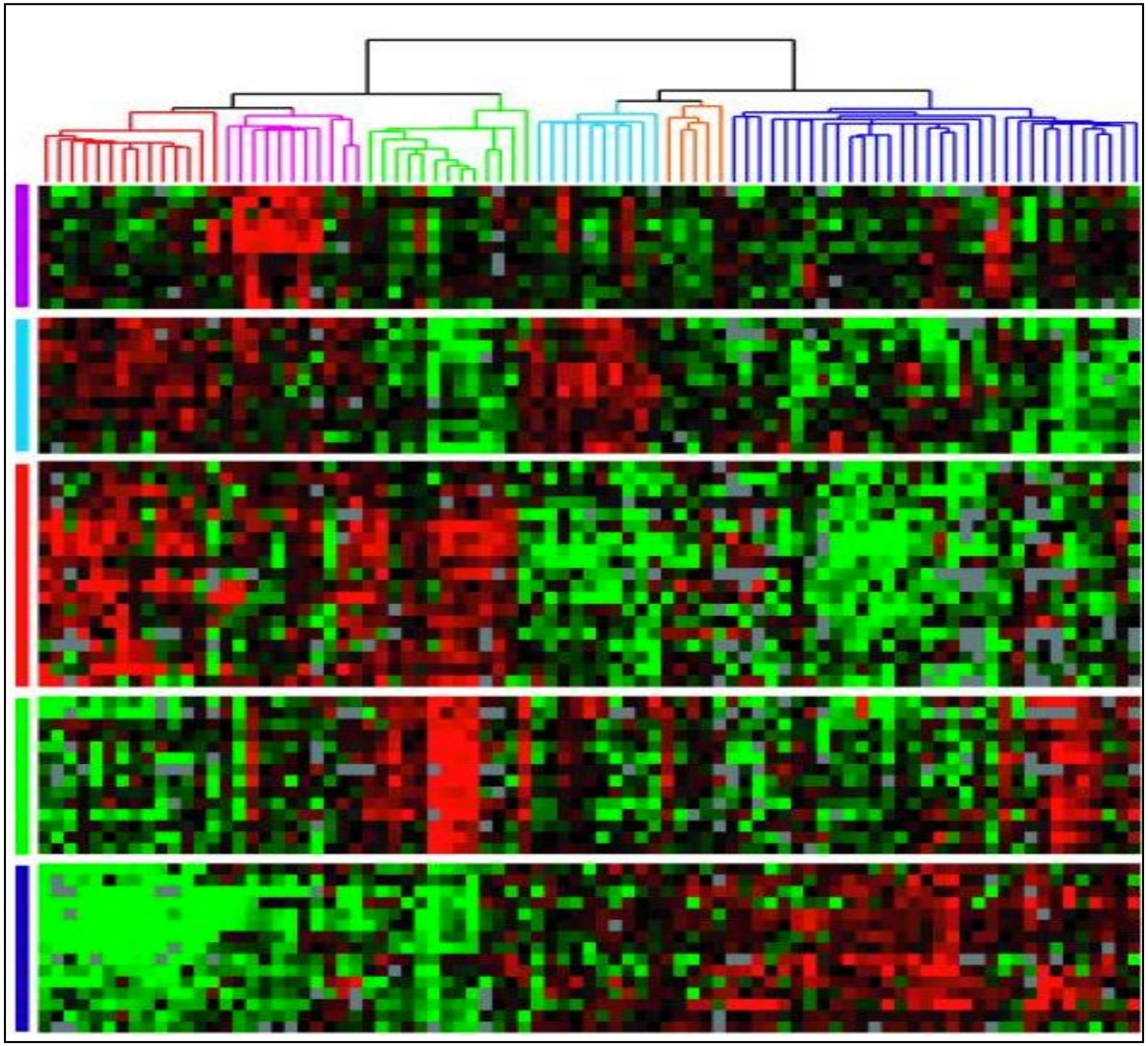
ChIP-seq



Hi-C

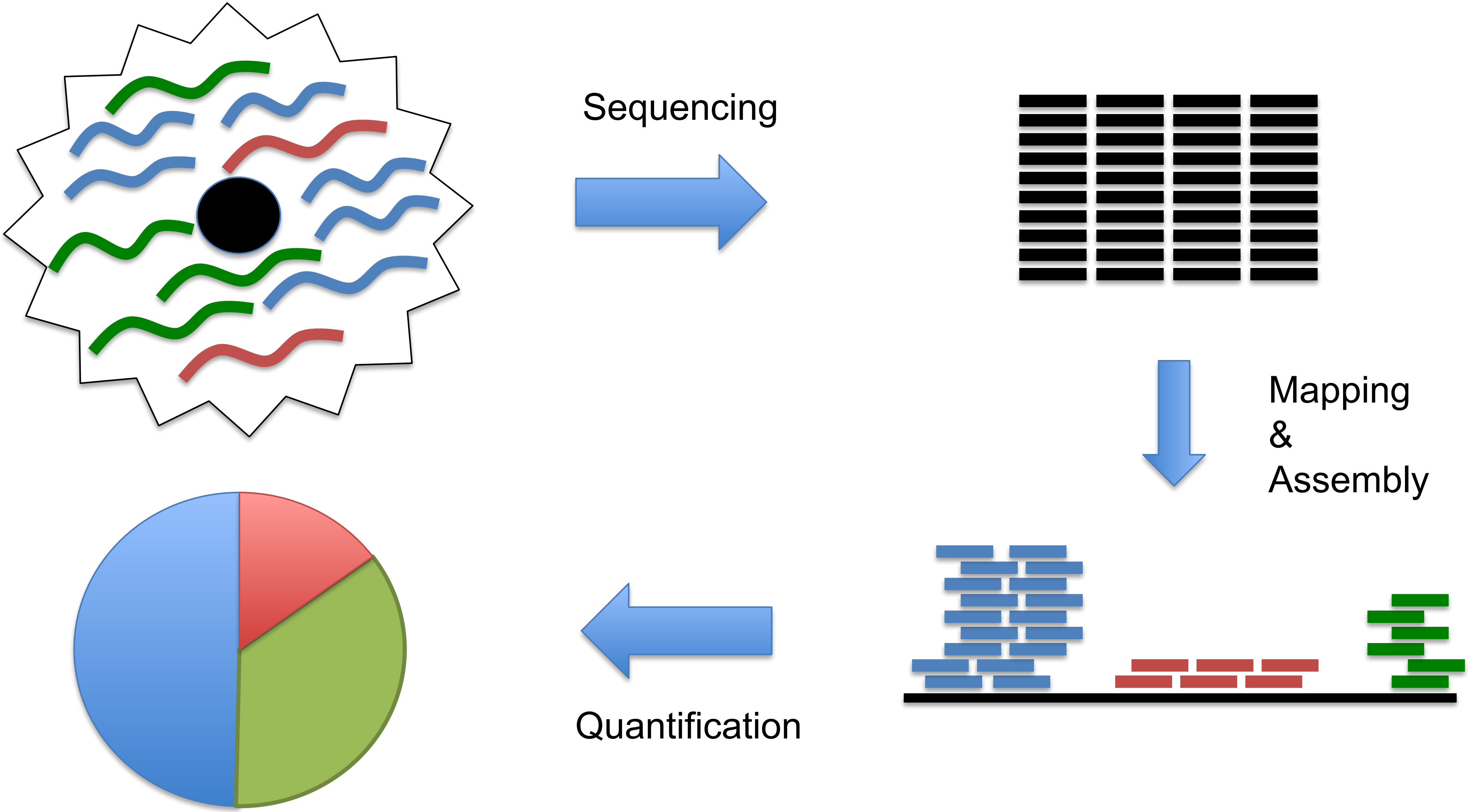


RNA-seq



Gene expression patterns of breast carcinomas distinguish tumor subclasses with clinical implications.
Sørlie et al (2001) *PNAS*. 98(19):10869-74.

RNA-seq Overview

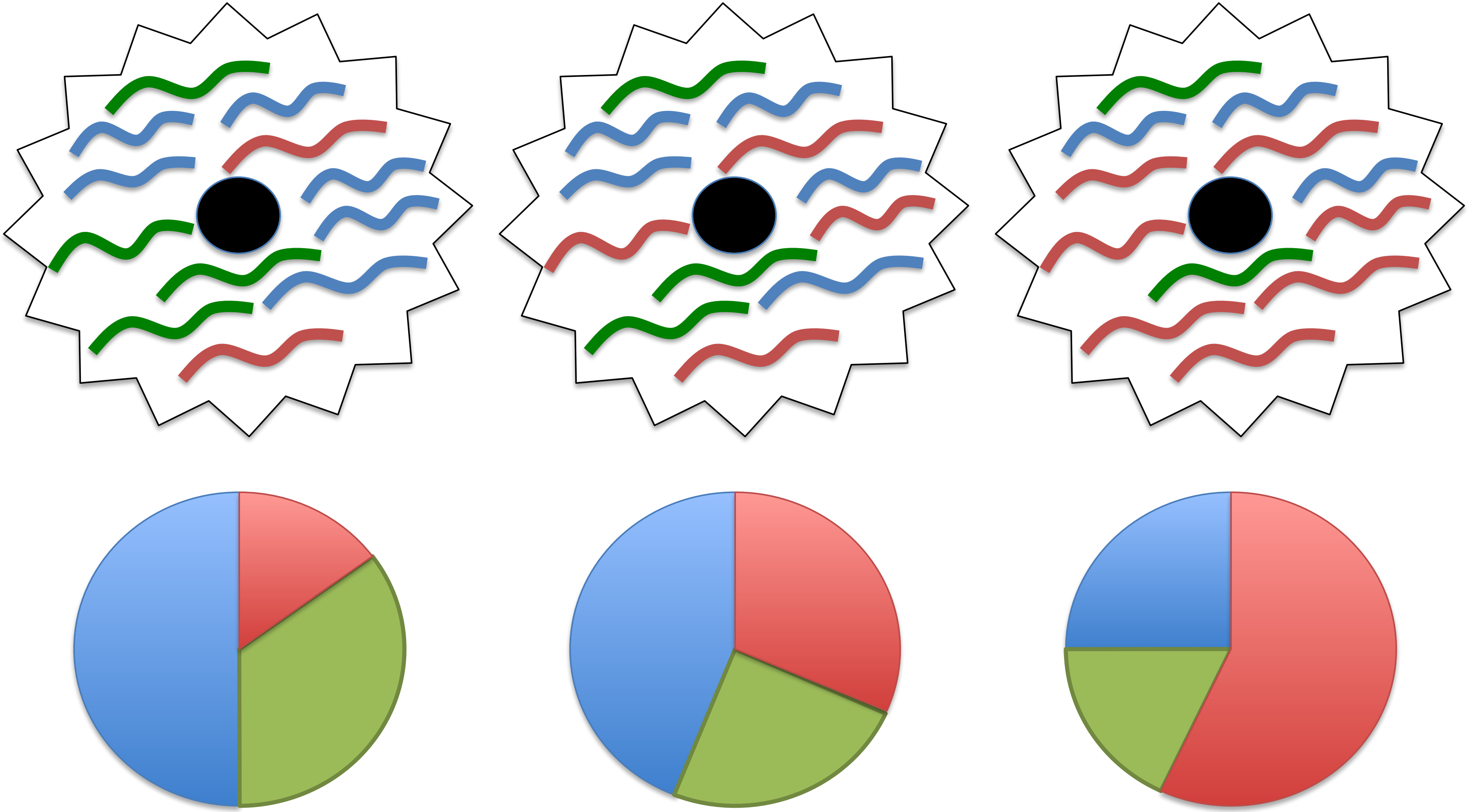


Sequencing

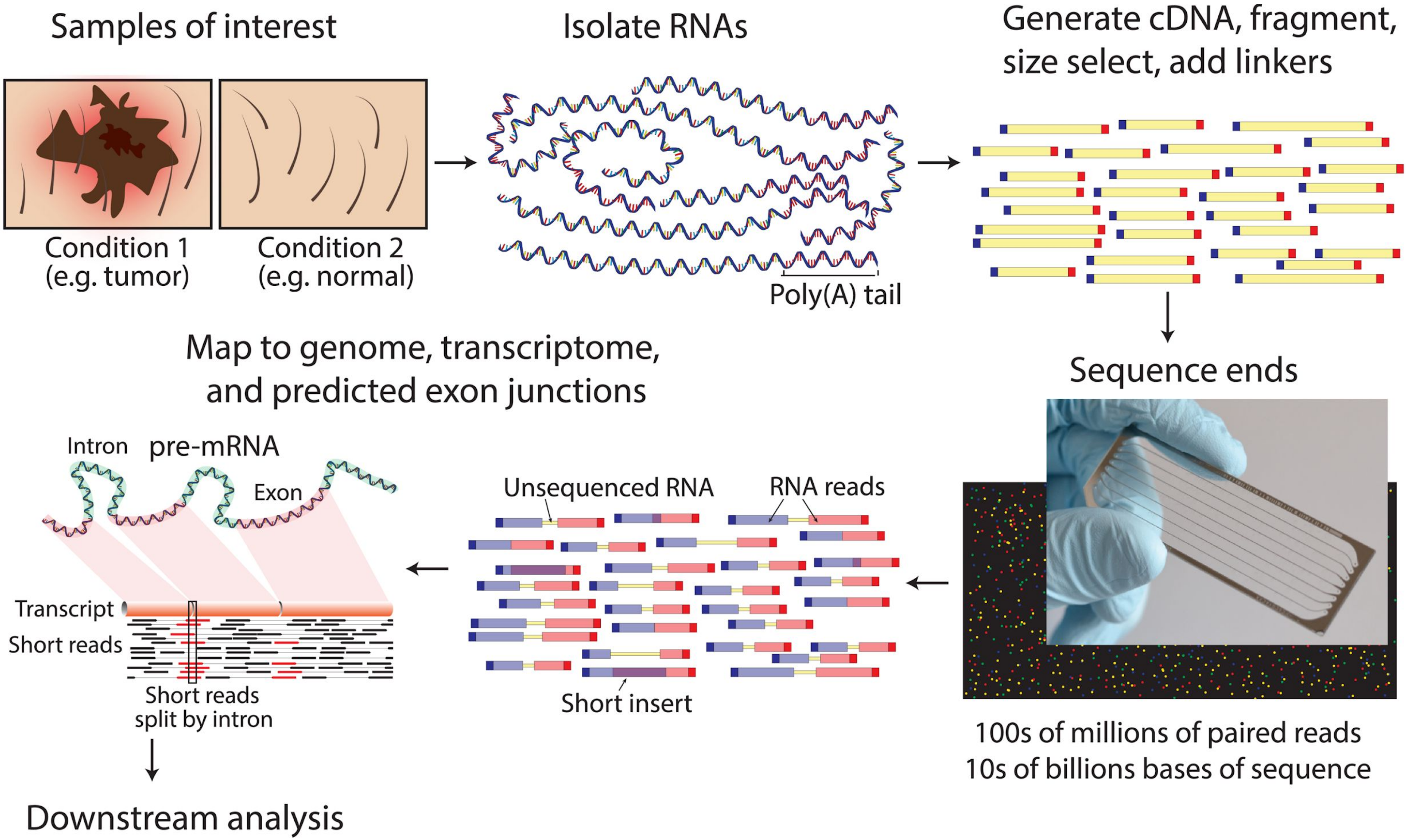
Mapping
&
Assembly

Quantification

RNA-seq Overview



RNA-seq Overview



Samples of interest

Condition 1
(e.g. tumor)

Condition 2
(e.g. normal)

Isolate RNAs

Poly(A) tail

Generate cDNA, fragment,
size select, add linkers

Sequence ends

100s of millions of paired reads
10s of billions bases of sequence

Unsequenced RNA

RNA reads

Short insert

Map to genome, transcriptome,
and predicted exon junctions

Intron

pre-mRNA

Exon

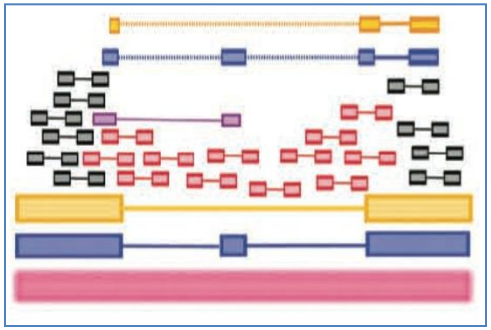
Transcript

Short reads

Short reads
split by intron

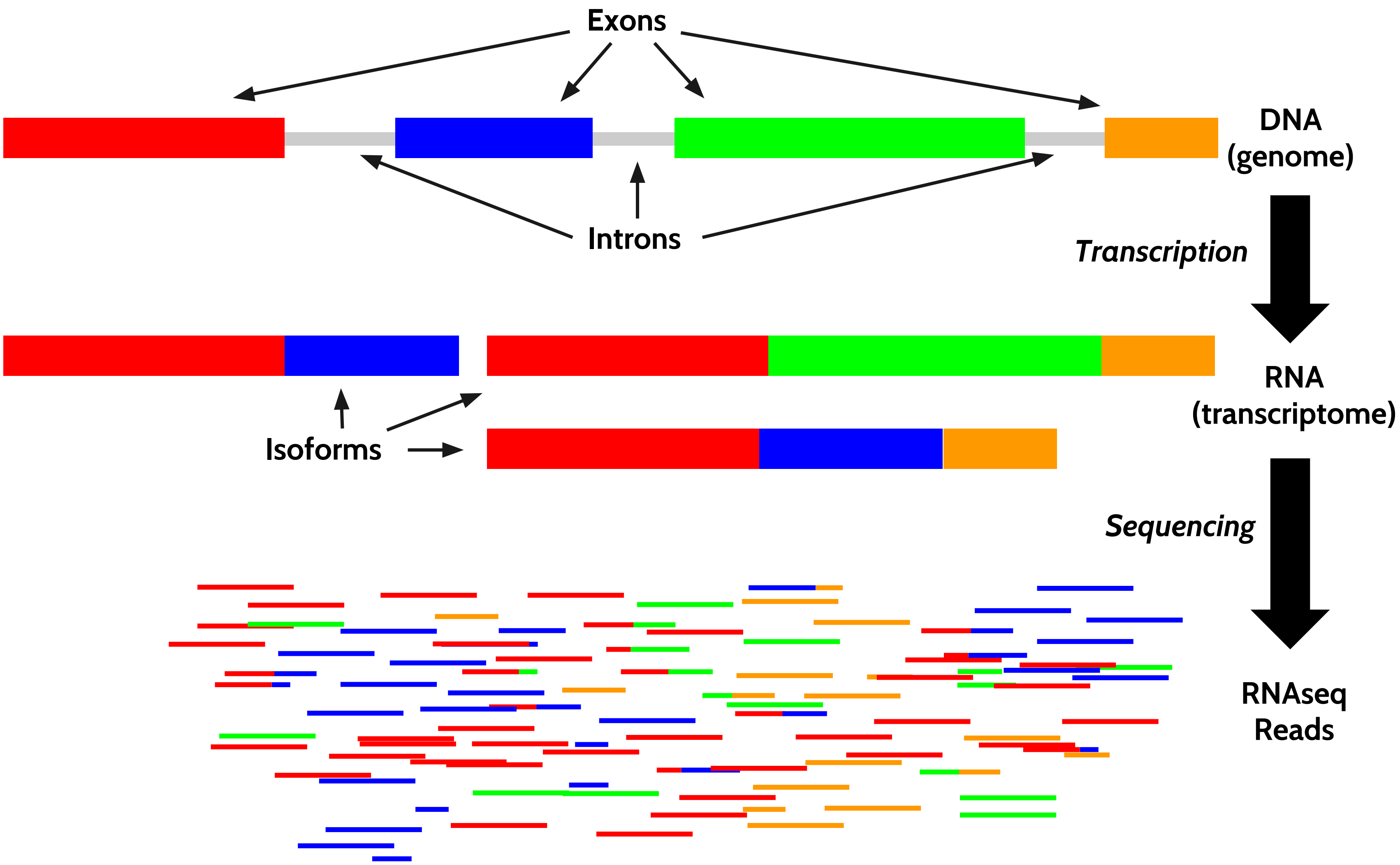
Downstream analysis

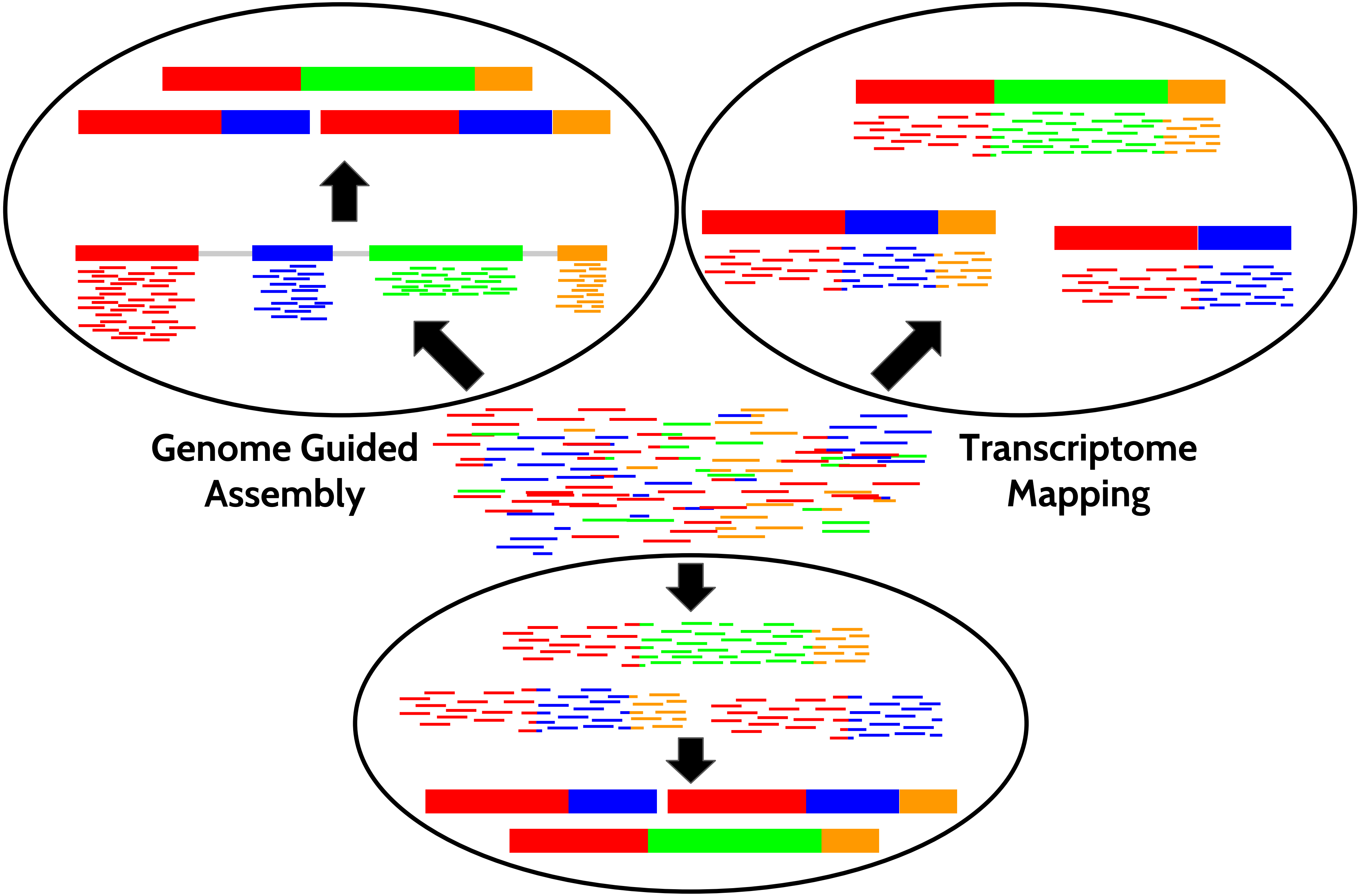
RNA-seq Challenges



Challenge 1: Eukaryotic genes are spliced

Alternative Splicing





Genome Guided Assembly

Transcriptome Mapping

De novo assembly

RNA-Seq Approaches

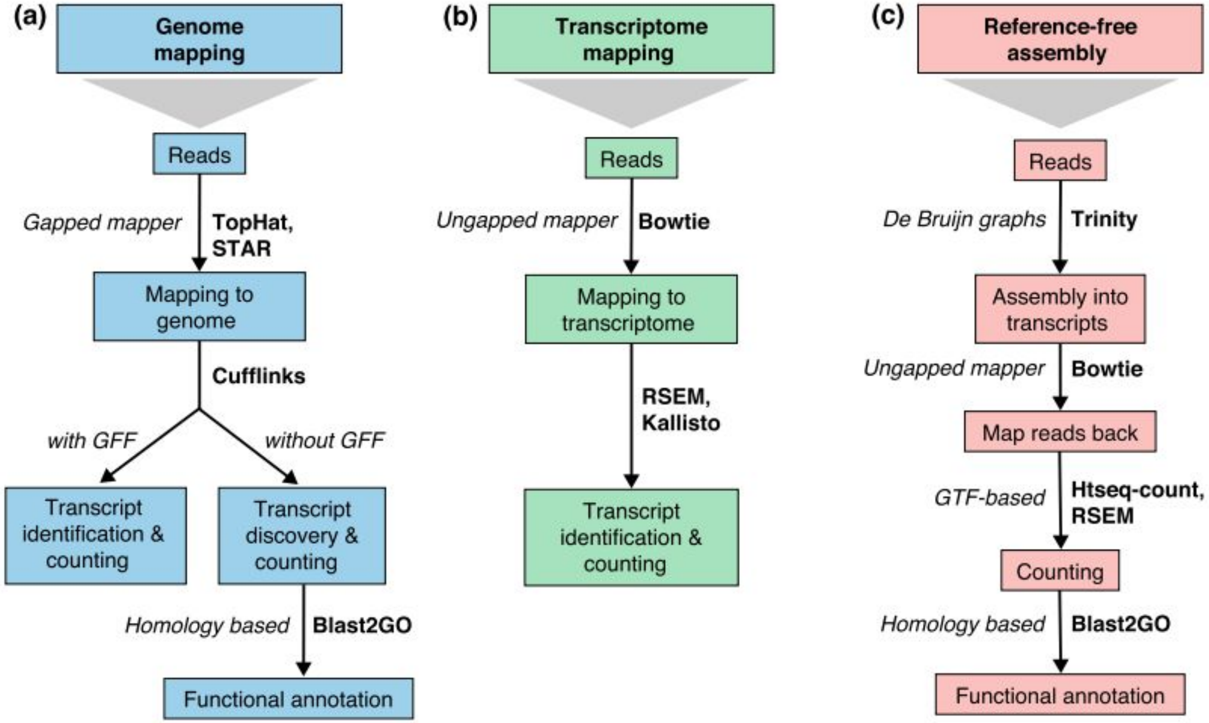


Fig. 2 Read mapping and transcript identification strategies. Three basic strategies for regular RNA-seq analysis. **a** An annotated genome is available and reads are mapped to the genome with a gapped mapper. Next (novel) transcript discovery and quantification can proceed with or without an annotation file. Novel transcripts are then functionally annotated. **b** If no novel transcript discovery is needed, reads can be mapped to the reference transcriptome using an ungapped aligner. Transcript identification and quantification can occur simultaneously. **c** When no genome is available, reads need to be assembled first into contigs or transcripts. For quantification, reads are mapped back to the novel reference transcriptome and further analysis proceeds as in (**b**) followed by the functional annotation of the novel transcripts as in (**a**). Representative software that can be used at each analysis step are indicated in *bold text*. Abbreviations: *GFF* General Feature Format, *GTF* gene transfer format, *RSEM* RNA-Seq by Expectation Maximization

A survey of best practices for RNA-seq data analysis
Conesa et al (2016) Genome Biology. doi 10.1186/s13059-016-0881-8

RNA-Seq Approaches

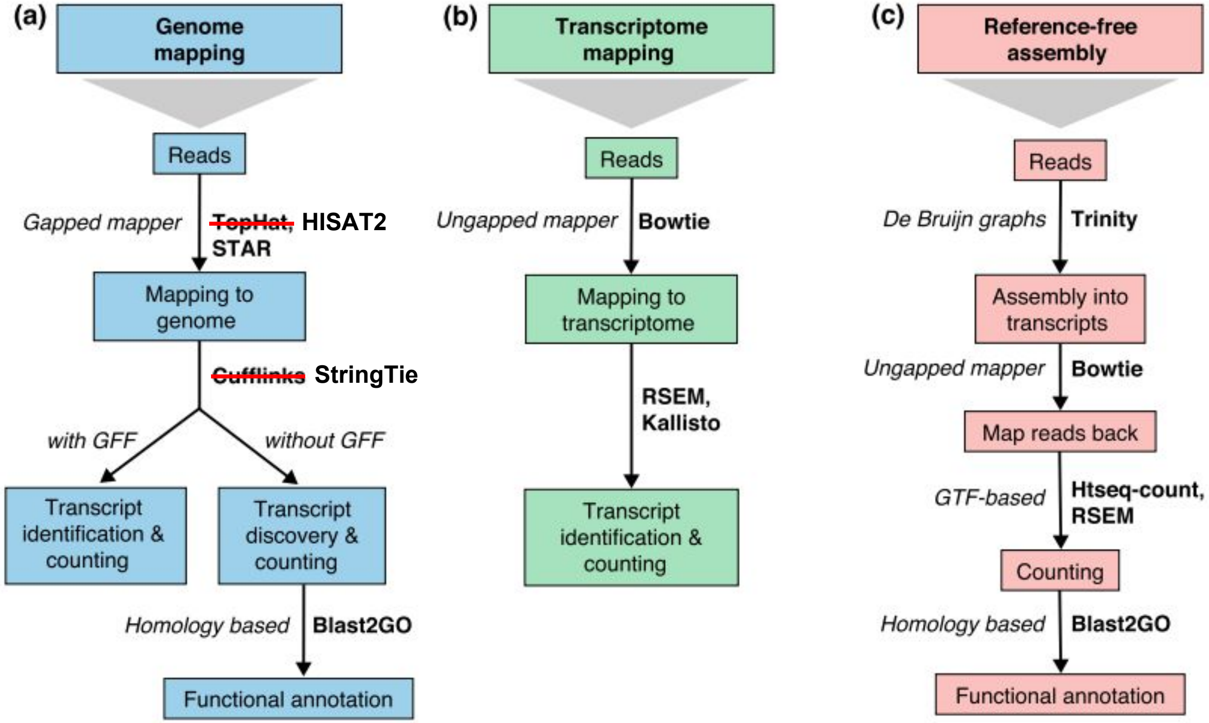


Fig. 2 Read mapping and transcript identification strategies. Three basic strategies for regular RNA-seq analysis. **a** An annotated genome is available and reads are mapped to the genome with a gapped mapper. Next (novel) transcript discovery and quantification can proceed with or without an annotation file. Novel transcripts are then functionally annotated. **b** If no novel transcript discovery is needed, reads can be mapped to the reference transcriptome using an ungapped aligner. Transcript identification and quantification can occur simultaneously. **c** When no genome is available, reads need to be assembled first into contigs or transcripts. For quantification, reads are mapped back to the novel reference transcriptome and further analysis proceeds as in (**b**) followed by the functional annotation of the novel transcripts as in (**a**). Representative software that can be used at each analysis step are indicated in *bold text*. Abbreviations: *GFF* General Feature Format, *GTF* gene transfer format, *RSEM* RNA-Seq by Expectation Maximization

A survey of best practices for RNA-seq data analysis
Conesa et al (2016) Genome Biology. doi 10.1186/s13059-016-0881-8

RNA-Seq Approaches

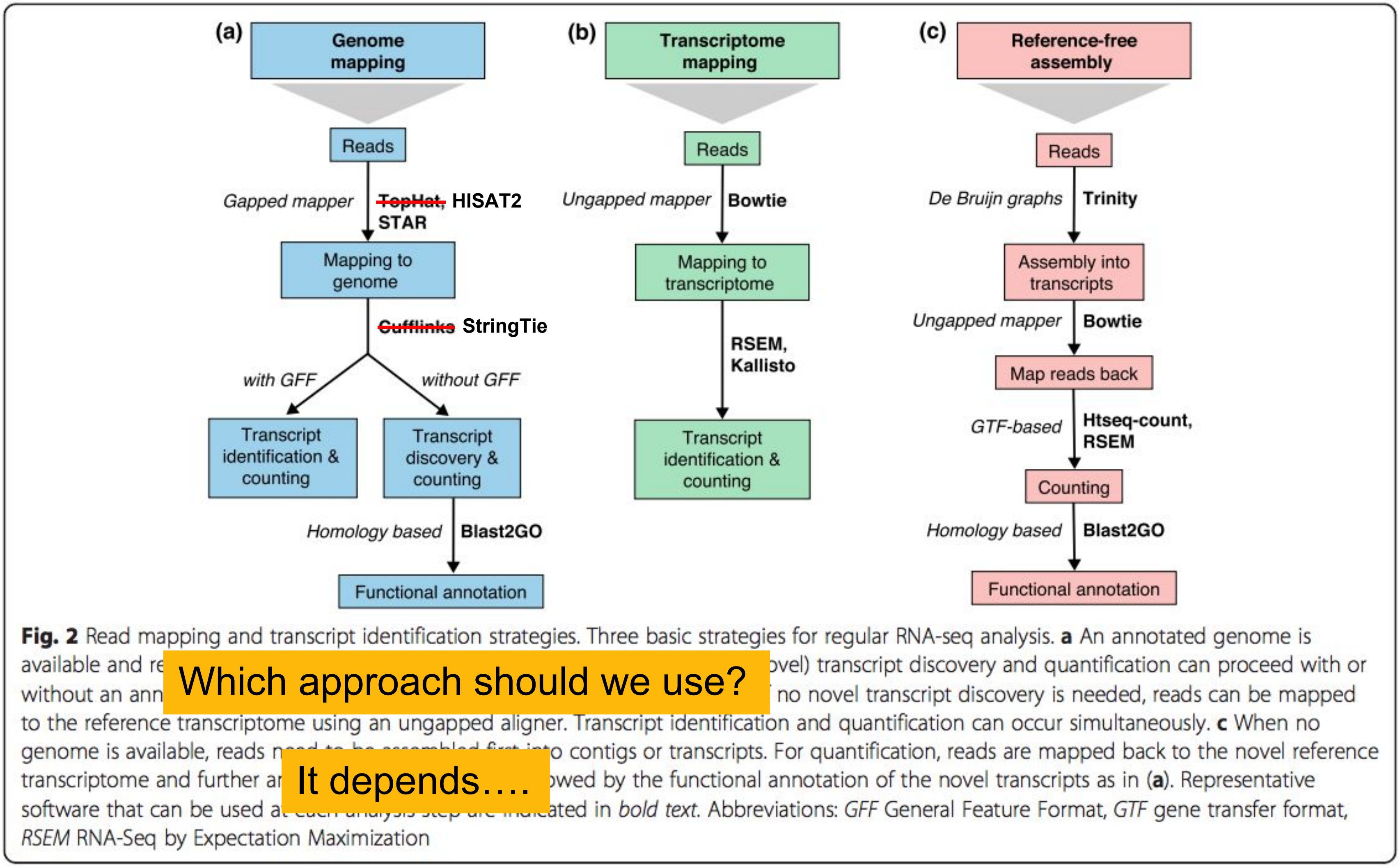


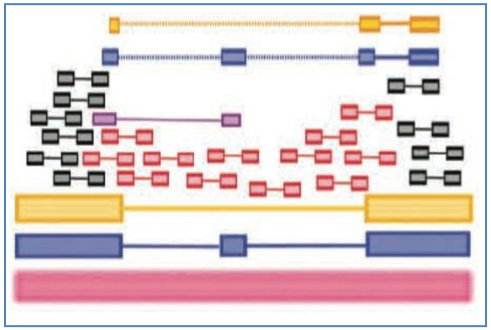
Fig. 2 Read mapping and transcript identification strategies. Three basic strategies for regular RNA-seq analysis. **a** An annotated genome is available and re vel) transcript discovery and quantification can proceed with or without an ann no novel transcript discovery is needed, reads can be mapped to the reference transcriptome using an ungapped aligner. Transcript identification and quantification can occur simultaneously. **c** When no genome is available, reads need to be assembled first into contigs or transcripts. For quantification, reads are mapped back to the novel reference transcriptome and further ar owed by the functional annotation of the novel transcripts as in (**a**). Representative software that can be used at each analysis step are indicated in *bold text*. Abbreviations: *GFF* General Feature Format, *GTF* gene transfer format, *RSEM* RNA-Seq by Expectation Maximization

Which approach should we use?

It depends….

A survey of best practices for RNA-seq data analysis
Conesa et al (2016) Genome Biology. doi 10.1186/s13059-016-0881-8

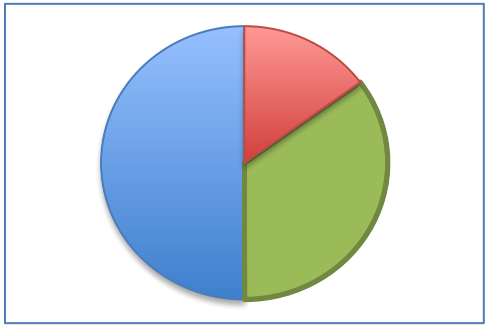
RNA-seq Challenges



Challenge 1: Eukaryotic genes are spliced

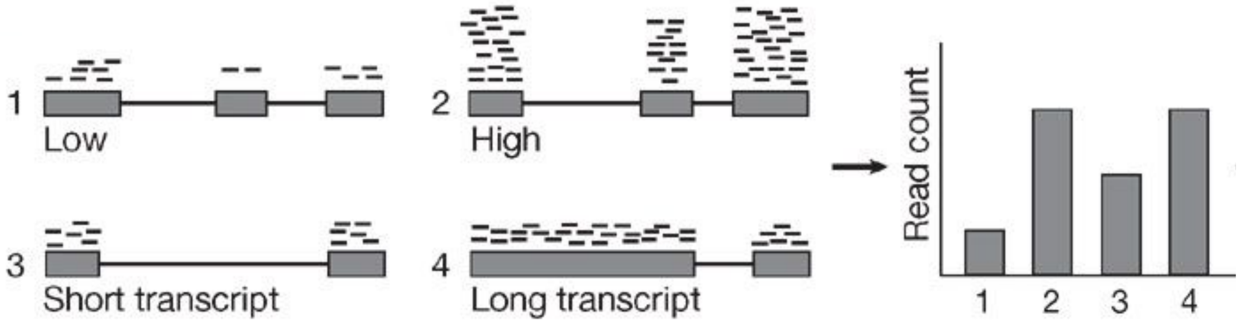
Solution: Use a spliced aligner, and assemble isoforms

TopHat: discovering spliced junctions with RNA-Seq.
Trapnell et al (2009) *Bioinformatics*. 25:0 1105-1111



Challenge 2: Read Count != Transcript abundance

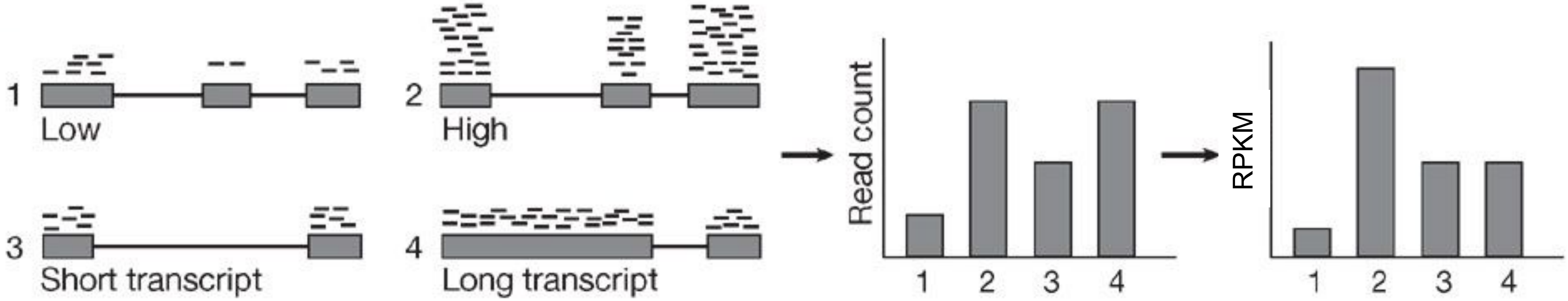
RPKM, FPKM, TPM



Counting Reads that align to a gene DOESN'T work!
- Overall Coverage: 1M reads in experiment 1 vs 10M reads in experiment 2
- Gene Length: gene 3 is 10kbp, gene 4 is 100kbp

1. RPKM: Reads Per Kilobase of Exon Per Million Reads Mapped (Mortazavi et al, 2008)

RPKM, FPKM, TPM



Counting Reads that align to a gene DOESN'T work!

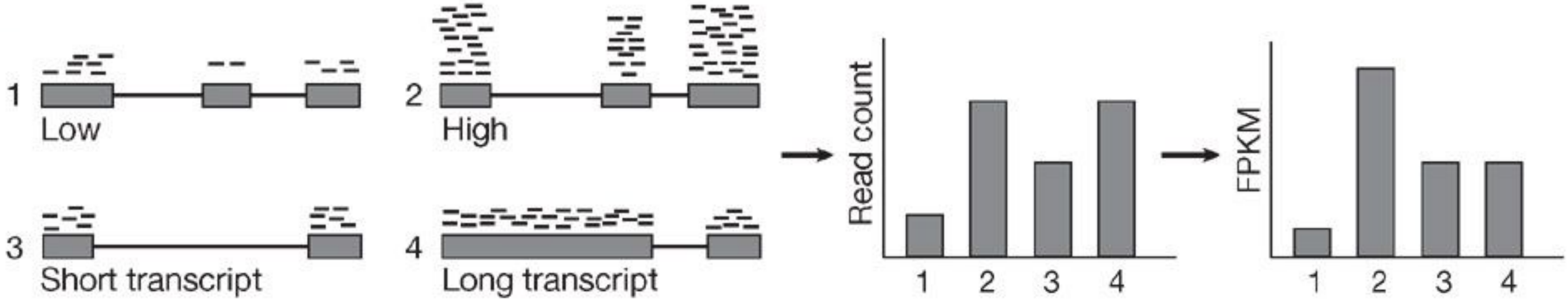
- Overall Coverage: 1M reads in experiment 1 vs 10M reads in experiment 2
- Gene Length: gene 3 is 10kbp, gene 4 is 100kbp

1. RPKM: Reads Per Kilobase of Exon Per Million Reads Mapped (Mortazavi et al, 2008)

(Count reads aligned to gene) / (length of gene in kilobases) / (# millions of read mapped)

=> Wait a second, reads in a pair arent independent!

RPKM, FPKM, TPM



Counting Reads that align to a gene DOESN'T work!

- Overall Coverage: 1M reads in experiment 1 vs 10M reads in experiment 2

- Gene Length: gene 3 is 10kbp, gene 4 is 100kbp

1. RPKM: Reads Per Kilobase of Exon Per Million Reads Mapped (Mortazavi et al, 2008)

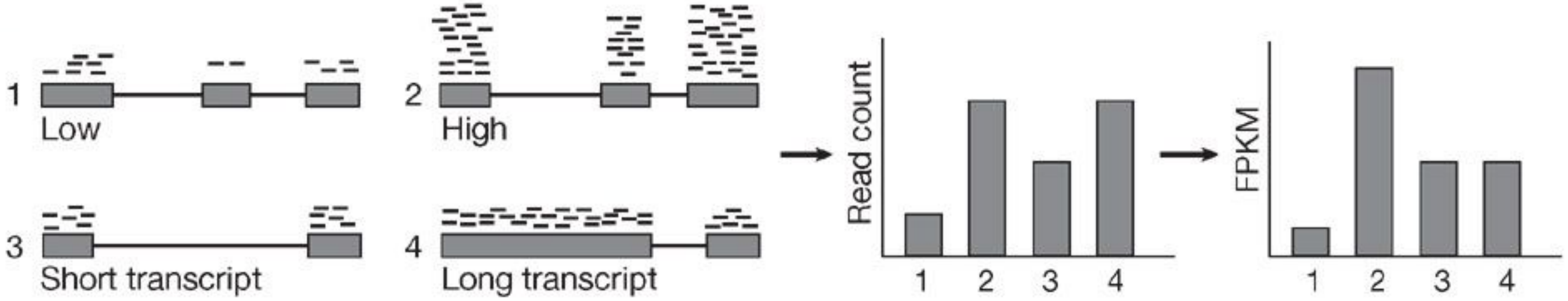
=> Wait a second, reads in a pair arent independent!

2. FPKM: Fragments Per Kilobase of Exon Per Million Reads Mapped (Trapnell et al, 2010)

⇒ Does a much better job with short exons & short genes by boosting coverage

⇒ Wait a second, FPKM depends on the average transcript length!

RPKM, FPKM, TPM



Counting Reads that align to a gene DOESN'T work!
- Overall Coverage: 1M reads in experiment 1 vs 10M reads in experiment 2
- Gene Length: gene 3 is 10kbp, gene 4 is 100kbp

1. RPKM: Reads Per Kilobase of Exon Per Million Reads Mapped (Mortazavi et al, 2008)
=> Wait a second, reads in a pair arent independent!

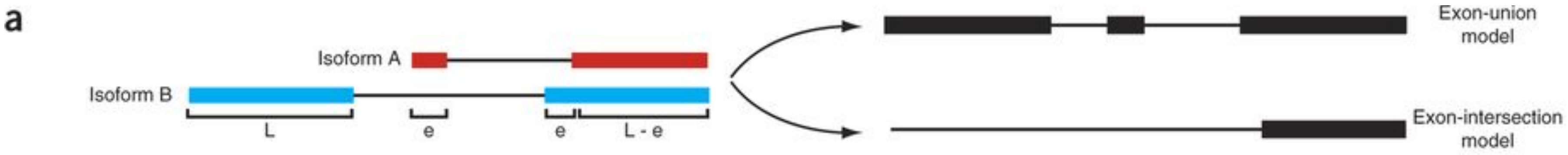
2. FPKM: Fragments Per Kilobase of Exon Per Million Reads Mapped (Trapnell et al, 2010)
=> Wait a second, FPKM depends on the average transcript length!

3. TPM: Transcripts Per Million (Li et al, 2011)
⇒ If you were to sequence one million full length transcripts, TPM is the number of transcripts you would have seen of type i, given the abundances of the other transcripts in your sample

=> Recommend you use TPM for all analysis, easy to compute given FPKM

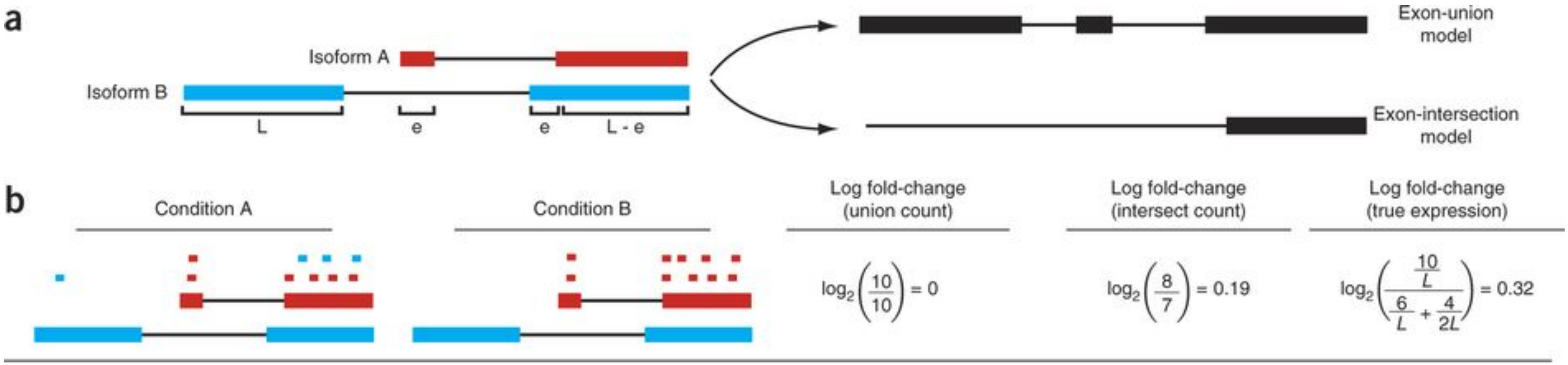
$$\mathrm{TPM}_i = \left(\frac{\mathrm{FPKM}_i}{\sum_j \mathrm{FPKM}_j}\right) \cdot 10^6$$

Gene or Isoform Quantification?



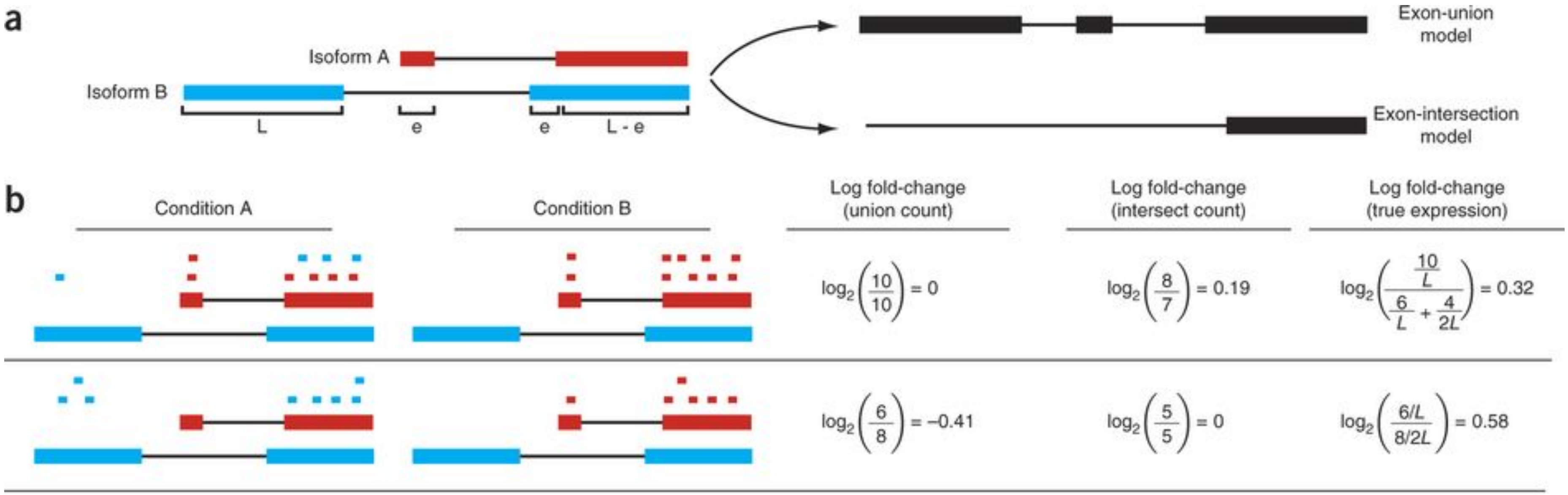
Differential analysis of gene regulation at transcript resolution with RNA-seq
Trapnell et al (2013) Nature Biotechnology 31, 46–53. doi:10.1038/nbt.2450

Gene or Isoform Quantification?


a
Isoform A
Isoform B
L
e
e
L - e
Exon-union model
Exon-intersection model
b
Condition A
Condition B
Log fold-change (union count)
Log fold-change (intersect count)
Log fold-change (true expression)
$\log_2\left(\frac{10}{10}\right) = 0$
$\log_2\left(\frac{8}{7}\right) = 0.19$
$\log_2\left(\frac{\frac{10}{L}}{\frac{6}{L}+\frac{4}{2L}}\right) = 0.32$

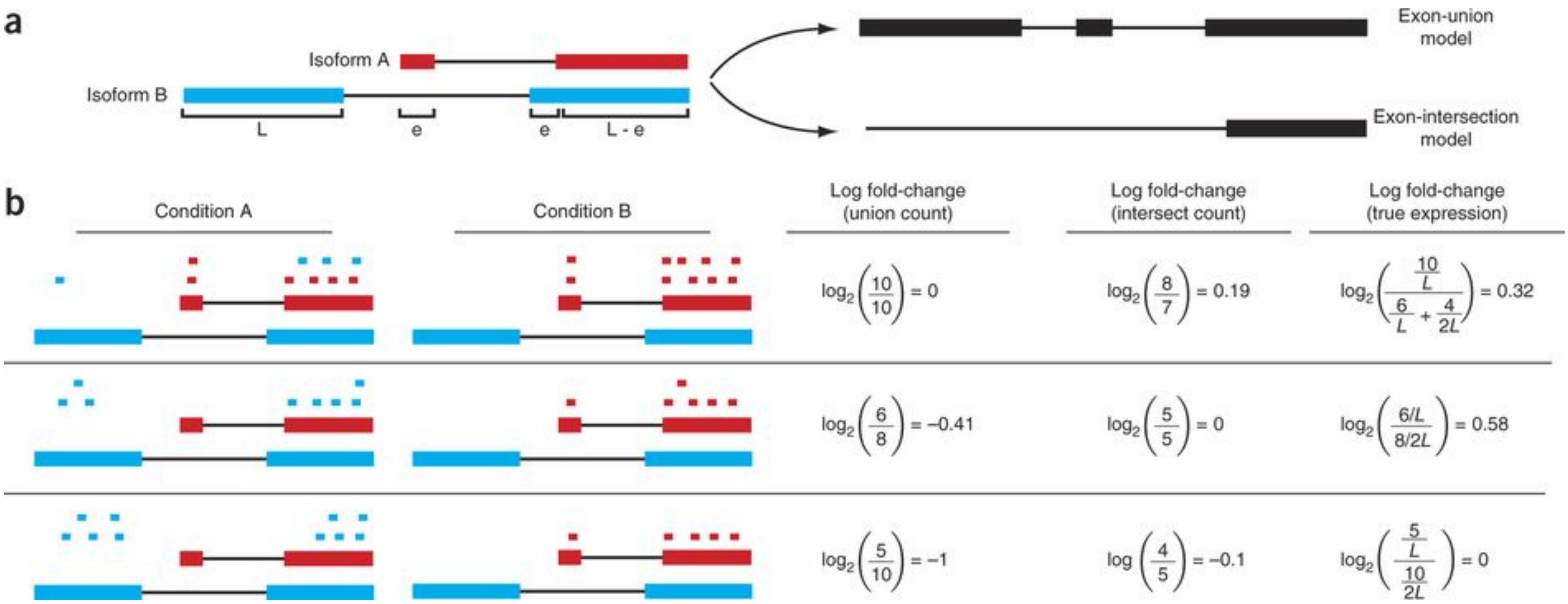
Differential analysis of gene regulation at transcript resolution with RNA-seq
Trapnell et al (2013) Nature Biotechnology 31, 46–53. doi:10.1038/nbt.2450

Gene or Isoform Quantification?


a
Isoform A
Isoform B
L
e
e
L - e
Exon-union model
Exon-intersection model
b
Condition A
Condition B
Log fold-change (union count)
Log fold-change (intersect count)
Log fold-change (true expression)
$\log_2\left(\frac{10}{10}\right) = 0$
$\log_2\left(\frac{8}{7}\right) = 0.19$
$\log_2\left(\frac{\frac{10}{L}}{\frac{6}{L}+\frac{4}{2L}}\right) = 0.32$
$\log_2\left(\frac{6}{8}\right) = -0.41$
$\log_2\left(\frac{5}{5}\right) = 0$
$\log_2\left(\frac{6/L}{8/2L}\right) = 0.58$

Differential analysis of gene regulation at transcript resolution with RNA-seq
Trapnell et al (2013) Nature Biotechnology 31, 46–53. doi:10.1038/nbt.2450

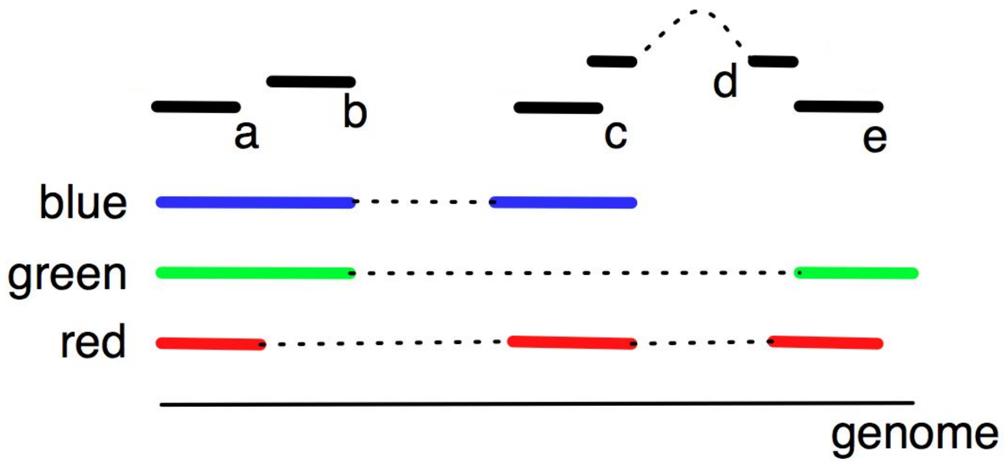
Gene or Isoform Quantification?


a
Isoform A
Isoform B
L
e
e
L - e
Exon-union model
Exon-intersection model
b
Condition A
Condition B
Log fold-change (union count)
Log fold-change (intersect count)
Log fold-change (true expression)
$\log_2\left(\frac{10}{10}\right) = 0$
$\log_2\left(\frac{8}{7}\right) = 0.19$
$\log_2\left(\frac{\frac{10}{L}}{\frac{6}{L}+\frac{4}{2L}}\right) = 0.32$
$\log_2\left(\frac{6}{8}\right) = -0.41$
$\log_2\left(\frac{5}{5}\right) = 0$
$\log_2\left(\frac{6/L}{8/2L}\right) = 0.58$
$\log_2\left(\frac{5}{10}\right) = -1$
$\log\left(\frac{4}{5}\right) = -0.1$
$\log_2\left(\frac{\frac{5}{L}}{\frac{10}{2L}}\right) = 0$

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

Differential analysis of gene regulation at transcript resolution with RNA-seq
Trapnell et al (2013) Nature Biotechnology 31, 46–53. doi:10.1038/nbt.2450

Multi-mapping? Isoform ambiguity? Expectation Maximization to the Rescue



The gene has three isoforms (red, green, blue) of the same length.
Our initial expectation is all 3 isoforms are equally expressed

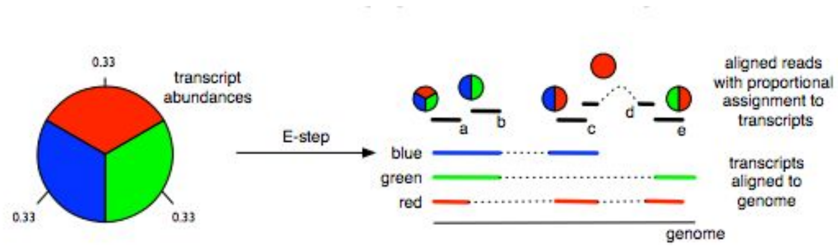
There are five reads (a,b,c,d,e) mapping to the gene.

- Read a maps to all three isoforms
- Read d only to red
- Reads b,c,e map to each of the three pairs of isoforms.

What is the most likely expression level of each isoform?

Models for transcript quantification from RNA-seq
Pachter, L (2011) arXiv. 1104.3889 [q-bio.GN]

Multi-mapping? Isoform ambiguity?
Expectation Maximization to the Rescue


0.33
transcript abundances
0.33
0.33
E-step
a
b
c
d
e
aligned reads with proportional assignment to transcripts
blue
green
red
transcripts aligned to genome
genome

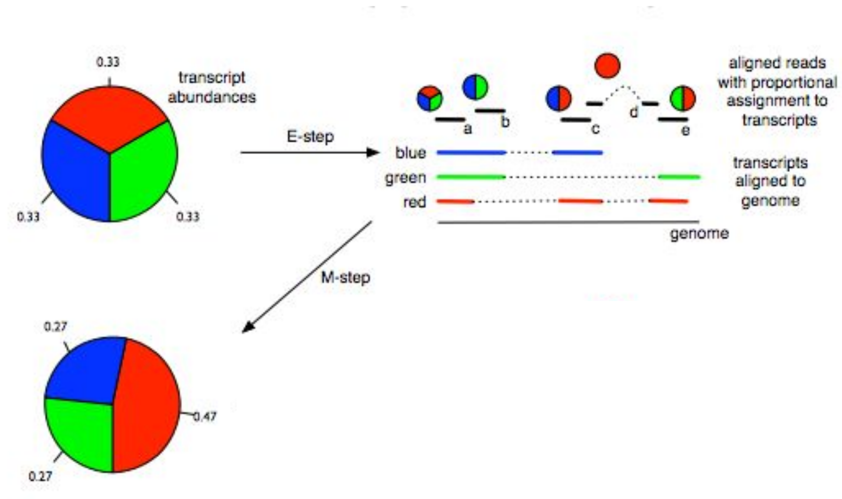
The gene has three isoforms (red, green, blue) of the same length. Initially every isoform is assigned the same abundance (red=1/3, green=1/3, blue=1/3)

There are five reads (a,b,c,d,e) mapping to the gene. Read a maps to all three isoforms, read d only to red, and the other three (reads b,c,e) to each of the three pairs of isoforms.

During the expectation (E) step reads are proportionately assigned to transcripts according to the (current) isoform abundances (RGB): a=(.33,.33,.33), b=(0,.5,.5), c=(.5,.5), d=(1,0,0), e=(.5,.5,0)

Models for transcript quantification from RNA-seq
Pachter, L (2011) arXiv. 1104.3889 [q-bio.GN]

Multi-mapping? Isoform ambiguity? Expectation Maximization to the Rescue


0.33
transcript abundances
0.33
0.33
E-step
aligned reads with proportional assignment to transcripts
a
b
c
d
e
blue
green
red
transcripts aligned to genome
genome
M-step
0.27
0.47
0.27

The gene has three isoforms (red, green, blue) of the same length. Initially every isoform is assigned the same abundance (red=1/3, green=1/3, blue=1/3)

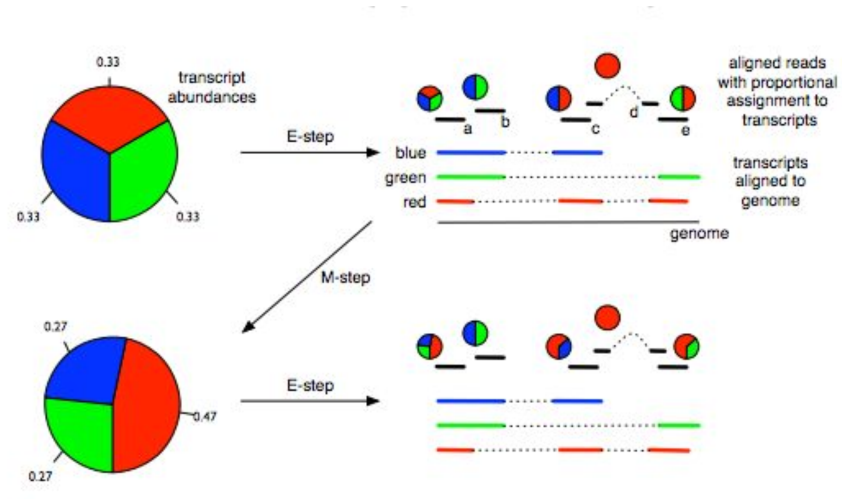
There are five reads (a,b,c,d,e) mapping to the gene. Read a maps to all three isoforms, read d only to red, and the other three (reads b,c,e) to each of the three pairs of isoforms.

During the expectation (E) step reads are proportionately assigned to transcripts according to the (current) isoform abundances (RGB): a=(.33,.33,.33), b=(0,.5,.5), c=(.5,.5), d=(1,0,0), e=(.5,.5,0)

Next, during the maximization (M) step isoform abundances are recalculated from the proportionately assigned read counts:
red: 0.47 = (0.33 + 0.5 + 1 + 0.5)/(2.33 + 1.33 + 1.33)
blue: 0.27 = (0.33 + 0.5 + 0.5)/(2.33 + 1.33 + 1.33)
green: 0.27 = (0.33 + 0.5 + 0.5)/(2.33 + 1.33 + 1.33)

Models for transcript quantification from RNA-seq
Pachter, L (2011) arXiv. 1104.3889 [q-bio.GN]

Multi-mapping? Isoform ambiguity? Expectation Maximization to the Rescue


0.33
transcript abundances
0.33
0.33
E-step
a
b
c
d
e
aligned reads with proportional assignment to transcripts
blue
green
red
transcripts aligned to genome
genome
M-step
0.27
0.47
0.27
E-step

The gene has three isoforms (red, green, blue) of the same length. Initially every isoform is assigned the same abundance (red=1/3, green=1/3, blue=1/3)

There are five reads (a,b,c,d,e) mapping to the gene. Read a maps to all three isoforms, read d only to red, and the other three (reads b,c,e) to each of the three pairs of isoforms.

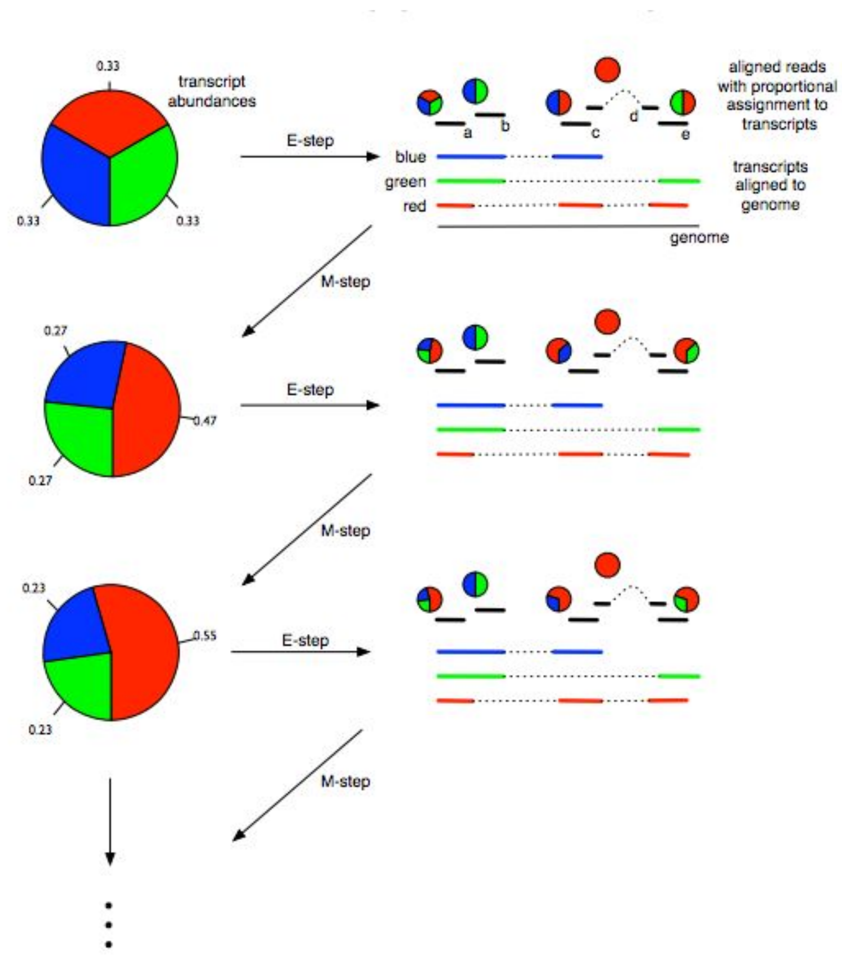
During the expectation (E) step reads are proportionately assigned to transcripts according to the (current) isoform abundances (RGB): a=(.33,.33,.33), b=(0,.5,.5), c=(.5,.5), d=(1,0,0), e=(.5,.5,0)

Next, during the maximization (M) step isoform abundances are recalculated from the proportionately assigned read counts:
red: 0.47 = (0.33 + 0.5 + 1 + 0.5)/(2.33 + 1.33 + 1.33)
blue: 0.27 = (0.33 + 0.5 + 0.5)/(2.33 + 1.33 + 1.33)
green: 0.27 = (0.33 + 0.5 + 0.5)/(2.33 + 1.33 + 1.33)

Repeat until convergence!

Models for transcript quantification from RNA-seq
Pachter, L (2011) arXiv. 1104.3889 [q-bio.GN]

Multi-mapping? Isoform ambiguity? Expectation Maximization to the Rescue



The gene has three isoforms (red, green, blue) of the same length. Initially every isoform is assigned the same abundance (red=1/3, green=1/3, blue=1/3)

There are five reads (a,b,c,d,e) mapping to the gene. Read a maps to all three isoforms, read d only to red, and the other three (reads b,c,e) to each of the three pairs of isoforms.

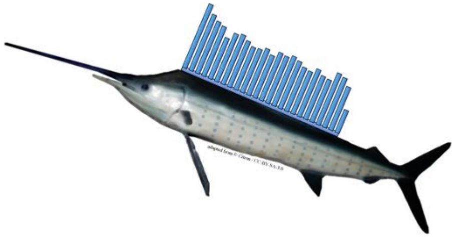
During the expectation (E) step reads are proportionately assigned to transcripts according to the (current) isoform abundances (RGB): a=(.33,.33,.33), b=(0,.5,.5), c=(.5,.5), d=(1,0,0), e=(.5,.5,0)

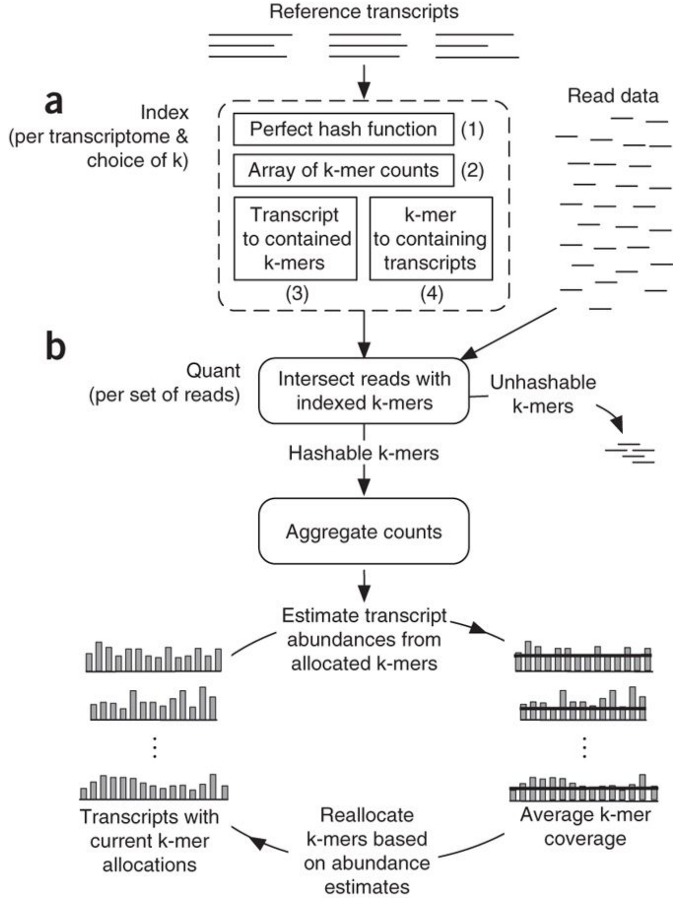
Next, during the maximization (M) step isoform abundances are recalculated from the proportionately assigned read counts:
red: 0.47 = (0.33 + 0.5 + 1 + 0.5)/(2.33 + 1.33 + 1.33)
blue: 0.27 = (0.33 + 0.5 + 0.5)/(2.33 + 1.33 + 1.33)
green: 0.27 = (0.33 + 0.5 + 0.5)/(2.33 + 1.33 + 1.33)

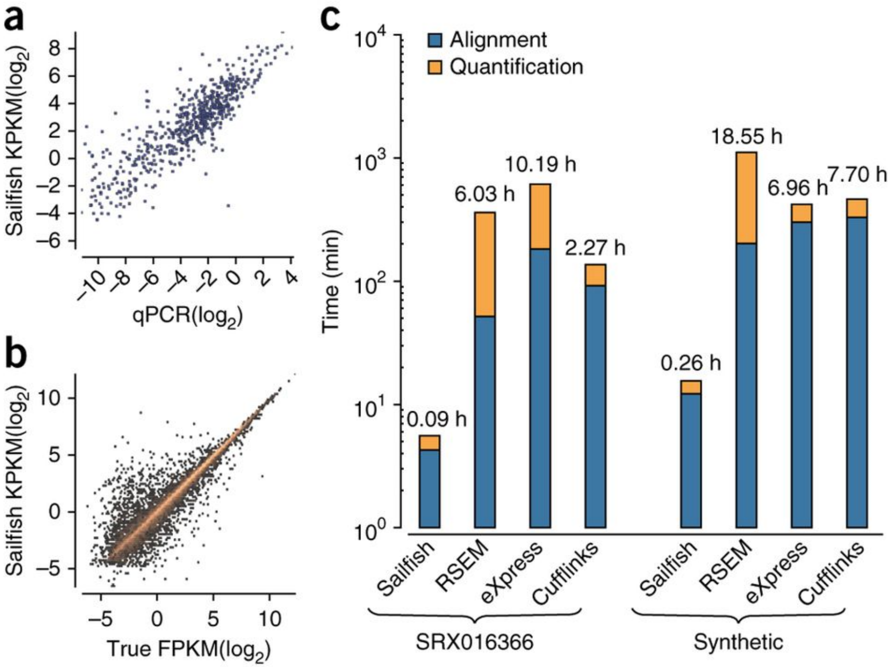
Repeat until convergence!

Models for transcript quantification from RNA-seq
Pachter, L (2011) arXiv. 1104.3889 [q-bio.GN]

Sailfish: Fast & Accurate RNA-seq Quantification




Reference transcripts
a
Index
(per transcriptome & choice of k)
Perfect hash function (1)
Array of k-mer counts (2)
Transcript to contained k-mers (3)
k-mer to containing transcripts (4)
Read data
b
Quant
(per set of reads)
Intersect reads with indexed k-mers
Unhashable k-mers
Hashable k-mers
Aggregate counts
Estimate transcript abundances from allocated k-mers
Average k-mer coverage
Reallocate k-mers based on abundance estimates
Transcripts with current k-mer allocations

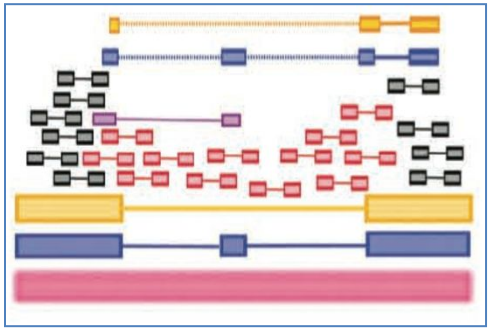


d

	Human brain tissue				Synthetic			
	Sailfish	RSEM	eXpress	Cufflinks	Sailfish	RSEM	eXpress	Cufflinks
Pearson	0.85	0.82	0.85	0.85	0.96	0.96	0.95	0.94
Spearman	0.84	0.80	0.85	0.85	0.76	0.77	0.77	0.75
RMSE	–	–	–	–	6.06	8.90	8.83	10.05
medPE	–	–	–	–	6.50	12.48	14.06	12.42

Sailfish enables alignment-free isoform quantification from RNA-seq reads using lightweight algorithms
Patro et al (2014) Nature Biotechnology 32, 462–464 doi:10.1038/nbt.2862

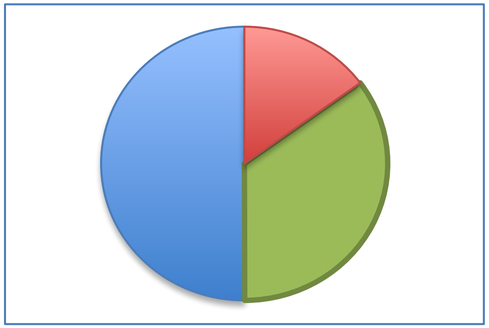
RNA-seq Challenges



Challenge 1: Eukaryotic genes are spliced

Solution: Use a spliced aligner, and assemble isoforms

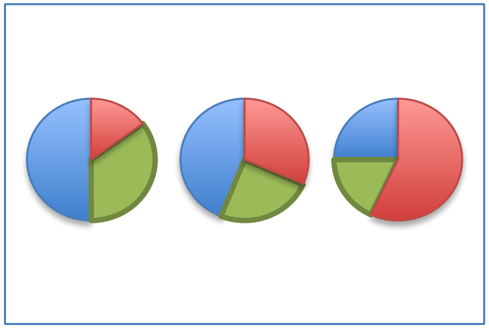
TopHat: discovering spliced junctions with RNA-Seq.
Trapnell et al (2009) *Bioinformatics.* 25:0 1105-1111



Challenge 2: Read Count != Transcript abundance

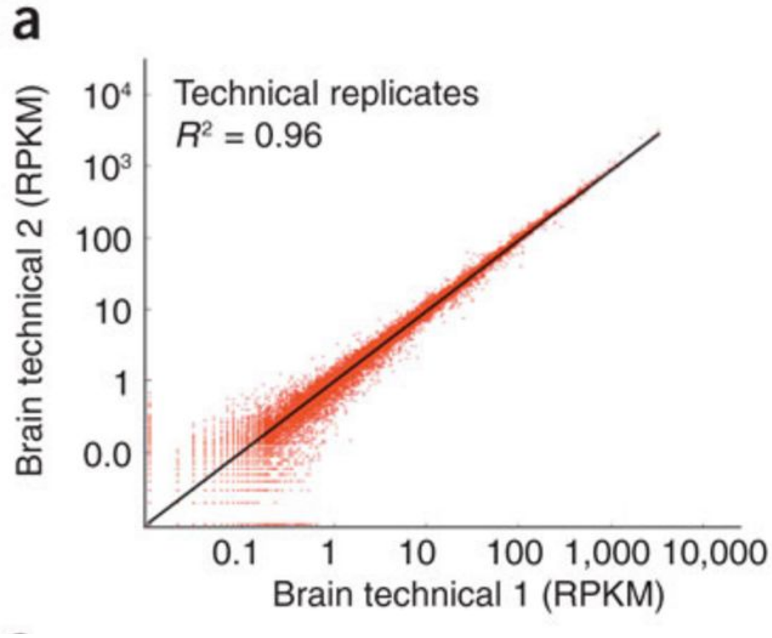
Solution: Infer underlying abundances (e.g. TPM)

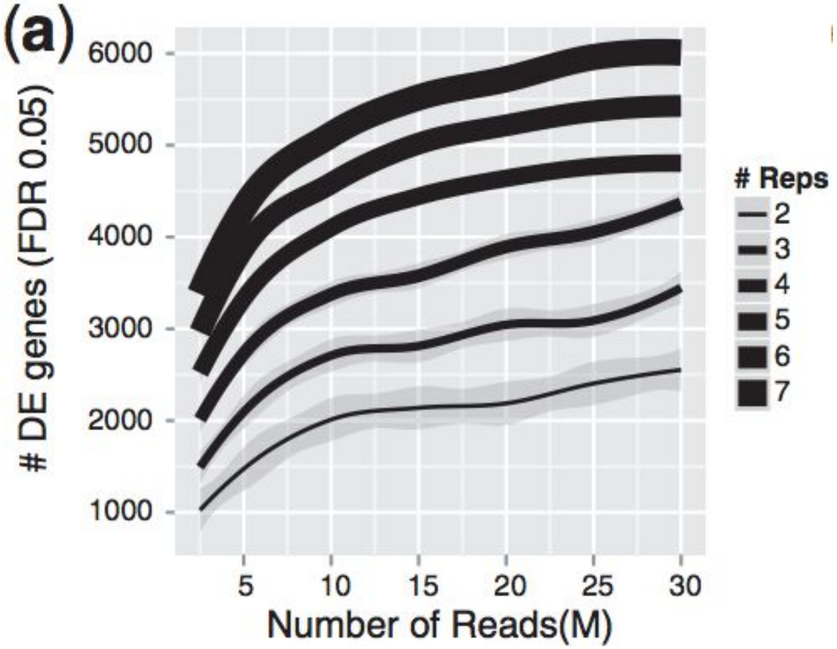
Transcript assembly and quantification by RNA-seq
Trapnell et al (2010) *Nat. Biotech.* 25(5): 511-515



Challenge 3: Transcript abundances are stochastic

How Many Replicates?



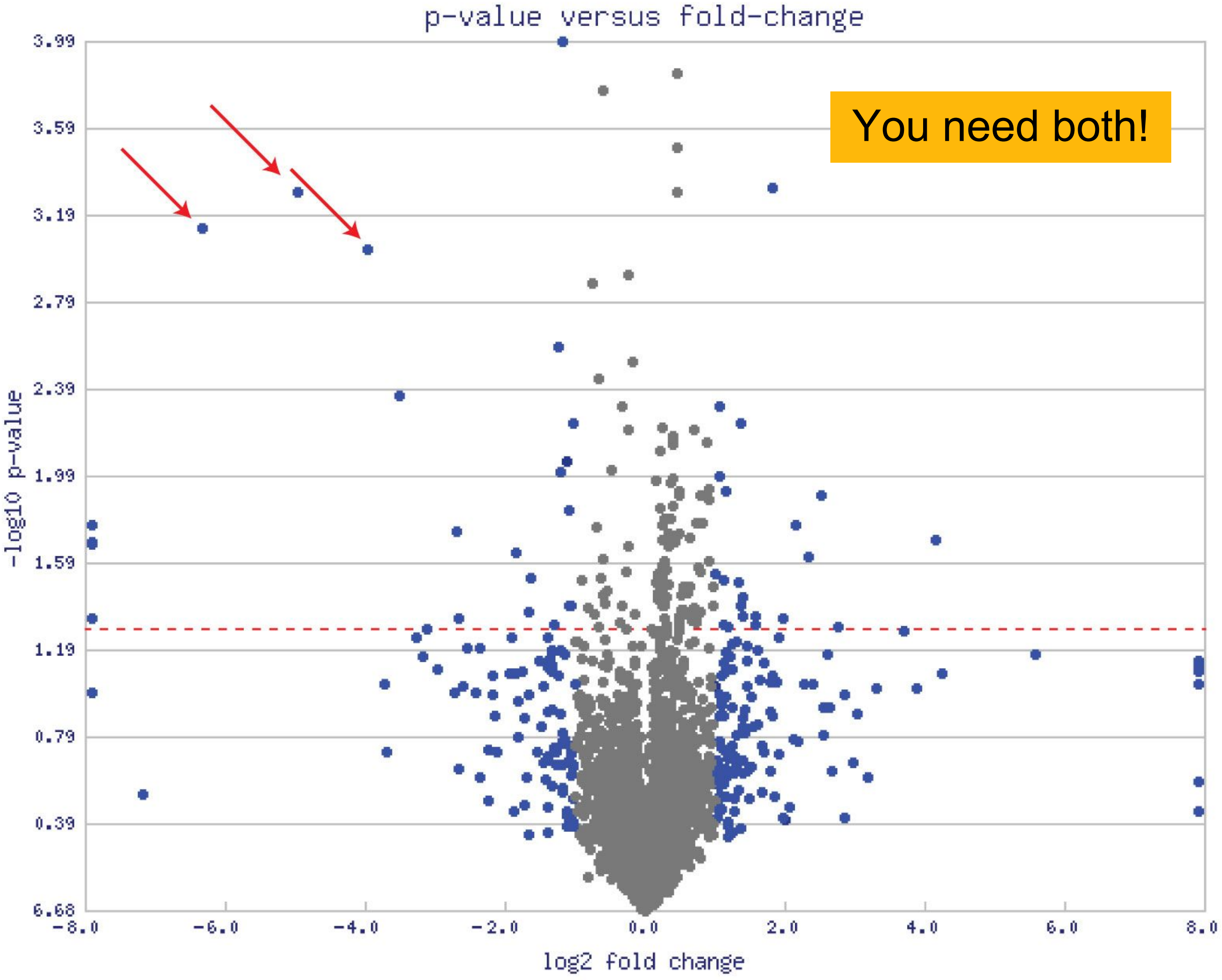


Why don’t we have perfect replicates?

Mapping and quantifying mammalian transcriptomes by RNA-Seq
Mortazavi et al (2008) Nature Methods. 5, 62-628

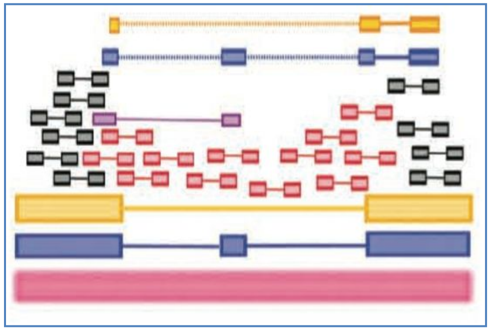
RNA-seq differential expression studies: more sequence or more replication?
Liu et al (2013) Bioinformatics. doi:10.1093/bioinformatics/btt688

Fold Change vs P-Value



You need both!

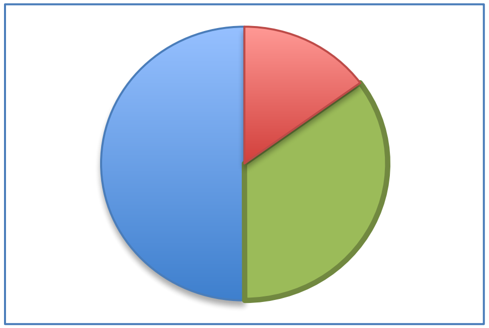
RNA-seq Challenges



Challenge 1: Eukaryotic genes are spliced

Solution: Use a spliced aligner, and assemble isoforms

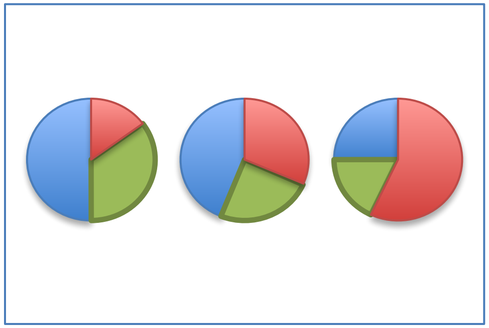
TopHat: discovering spliced junctions with RNA-Seq.
Trapnell et al (2009) *Bioinformatics.* 25:0 1105-1111



Challenge 2: Read Count != Transcript abundance

Solution: Infer underlying abundances (e.g. TPM)

Transcript assembly and quantification by RNA-seq
Trapnell et al (2010) *Nat. Biotech.* 25(5): 511-515

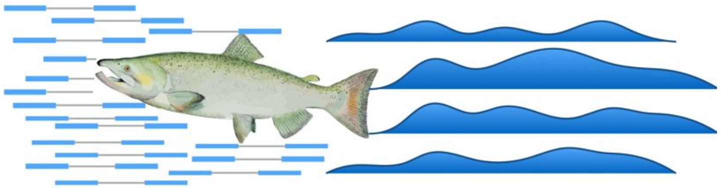


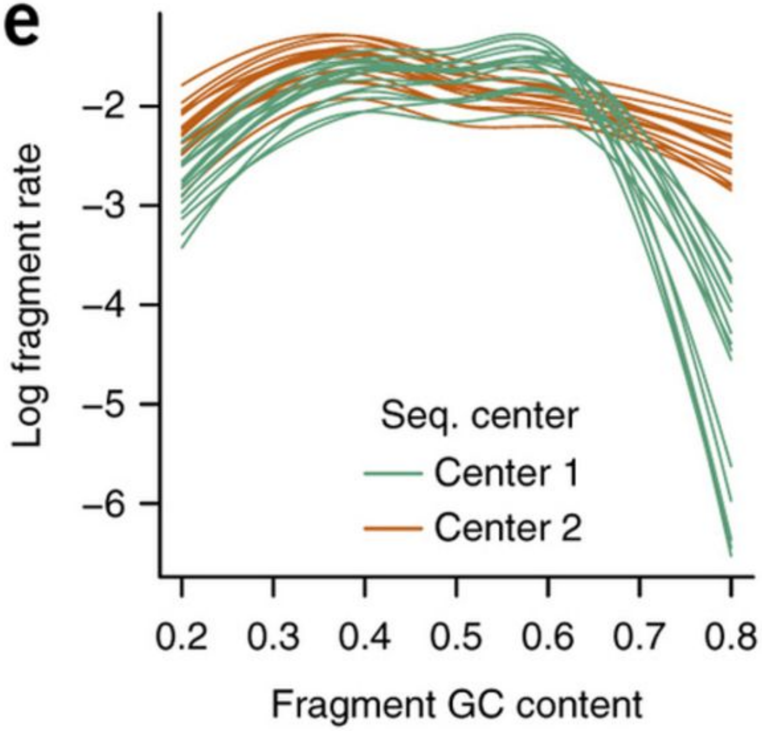
Challenge 3: Transcript abundances are stochastic

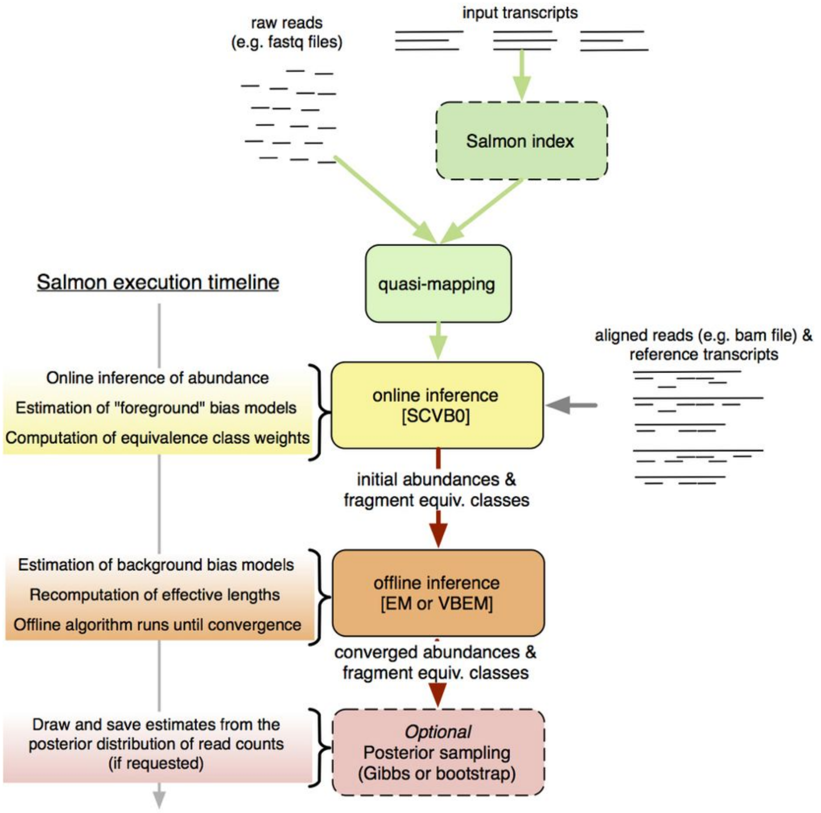
Solution: Replicates, replicates, and more replicates

RNA-seq differential expression studies: more sequence or more replication?
Liu et al (2013) *Bioinformatics.* doi:10.1093/bioinformatics/btt688

Salmon: The ultimate Quantification Pipeline?







raw reads (e.g. fastq files)

input transcripts

Salmon index

quasi-mapping

Salmon execution timeline

aligned reads (e.g. bam file) & reference transcripts

Online inference of abundance
Estimation of "foreground" bias models
Computation of equivalence class weights

online inference [SCVB0]

initial abundances & fragment equiv. classes

Estimation of background bias models
Recomputation of effective lengths
Offline algorithm runs until convergence

offline inference [EM or VBEM]

converged abundances & fragment equiv. classes

Draw and save estimates from the posterior distribution of read counts (if requested)

Optional
Posterior sampling (Gibbs or bootstrap)

Modeling of RNA-seq fragment sequence bias reduces systematic errors in transcript abundance estimation
Love et al (2016) Nature Biotechnology 34, 1287–1291 (2016) doi:10.1038/nbt.3682

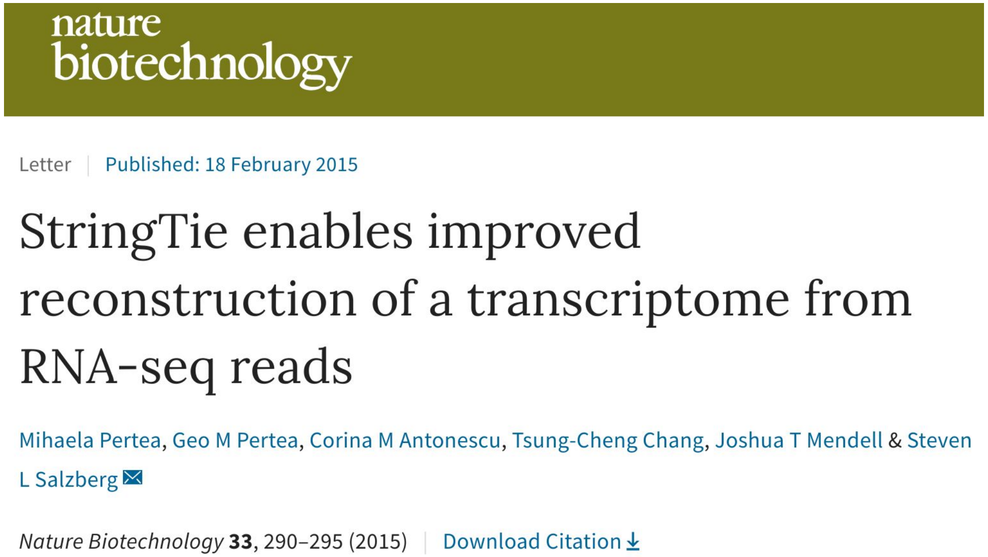
Salmon provides fast and bias-aware quantification of transcript expression
Patro et al (2017) Nature Methods (2017) doi:10.1038/nmeth.4197

Genome Guided Transcriptome Assembly

Most accurate high-throughput method for novel isoform discovery

Can also be guided by annotation, and produce quantification estimates

StringTie (JHU) and Scallop (CMU) are current state of the art



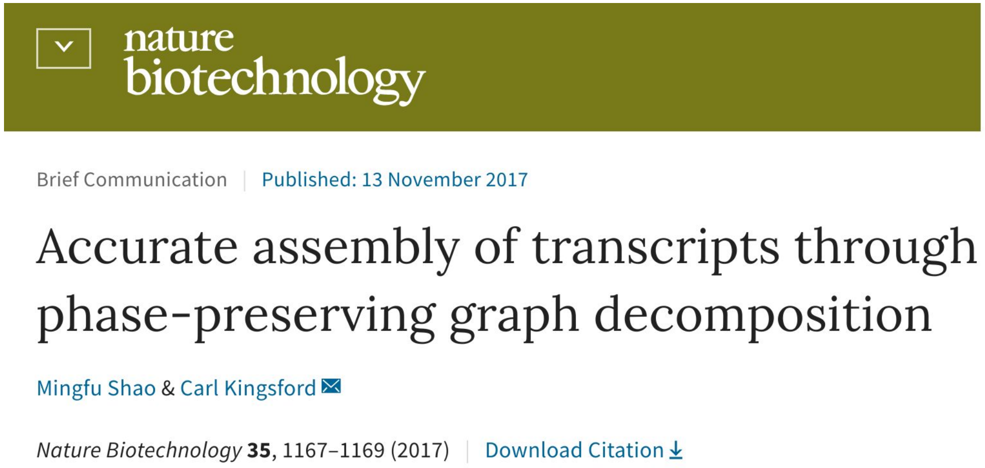
nature biotechnology

Letter | Published: 18 February 2015

StringTie enables improved reconstruction of a transcriptome from RNA-seq reads

Mihaela Pertea, Geo M Pertea, Corina M Antonescu, Tsung-Cheng Chang, Joshua T Mendell & Steven L Salzberg ✉

Nature Biotechnology **33**, 290–295 (2015) | Download Citation ⤓



nature biotechnology

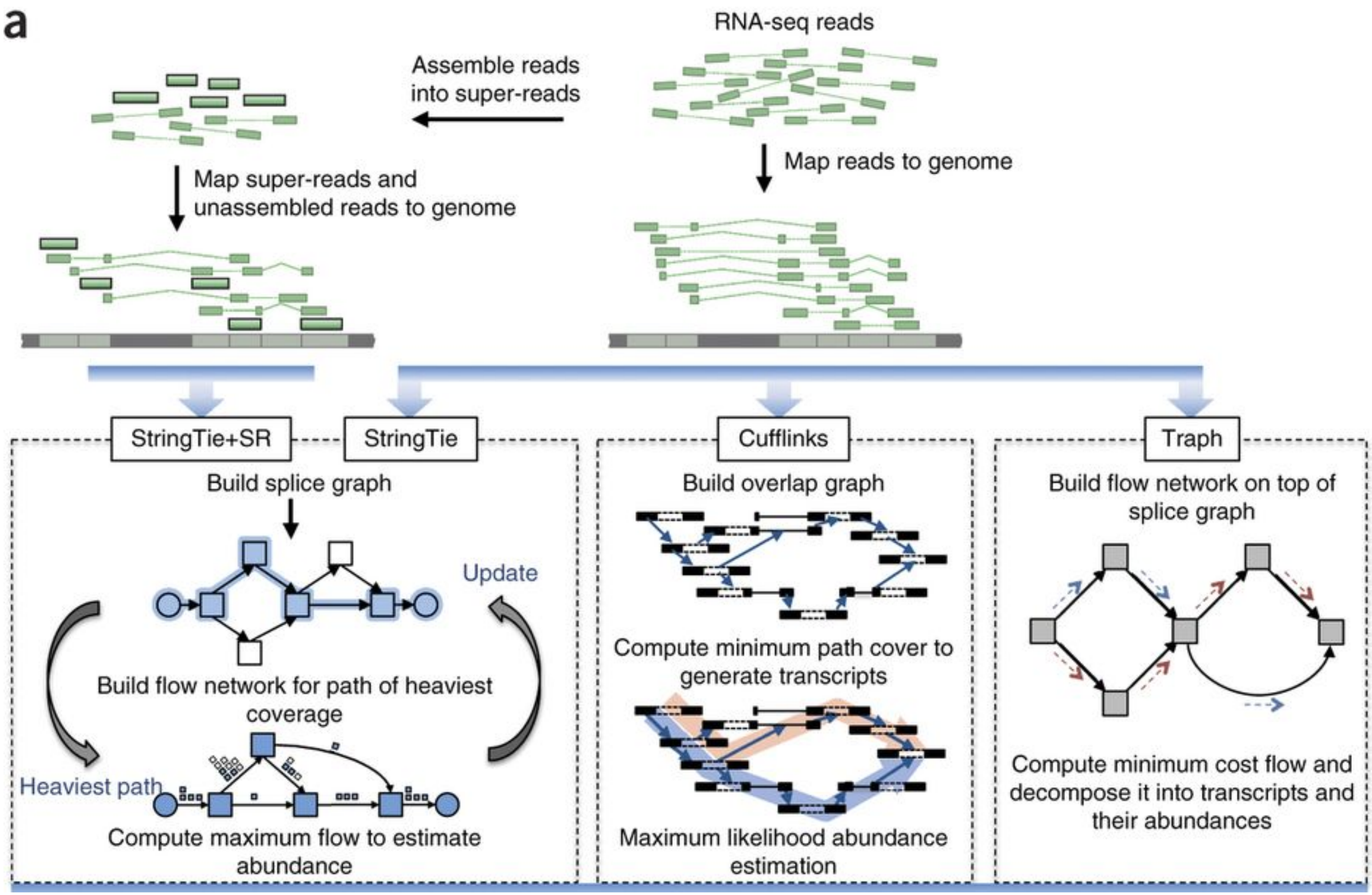
Brief Communication | Published: 13 November 2017

Accurate assembly of transcripts through phase-preserving graph decomposition

Mingfu Shao & Carl Kingsford ✉

Nature Biotechnology **35**, 1167–1169 (2017) | Download Citation ⤓

StringTie Algorithm


a
RNA-seq reads
Assemble reads into super-reads
Map reads to genome
Map super-reads and unassembled reads to genome
StringTie+SR
StringTie
Cufflinks
Traph
Build splice graph
Update
Build flow network for path of heaviest coverage
Heaviest path
Compute maximum flow to estimate abundance
Build overlap graph
Compute minimum path cover to generate transcripts
Maximum likelihood abundance estimation
Build flow network on top of splice graph
Compute minimum cost flow and decompose it into transcripts and their abundances

StringTie enables improved reconstruction of a transcriptome from RNA-seq reads.
Pertea M, et al. (2015) Nature Biotechnology. doi: 10.1038/nbt.3122.

Long-Read RNAseq

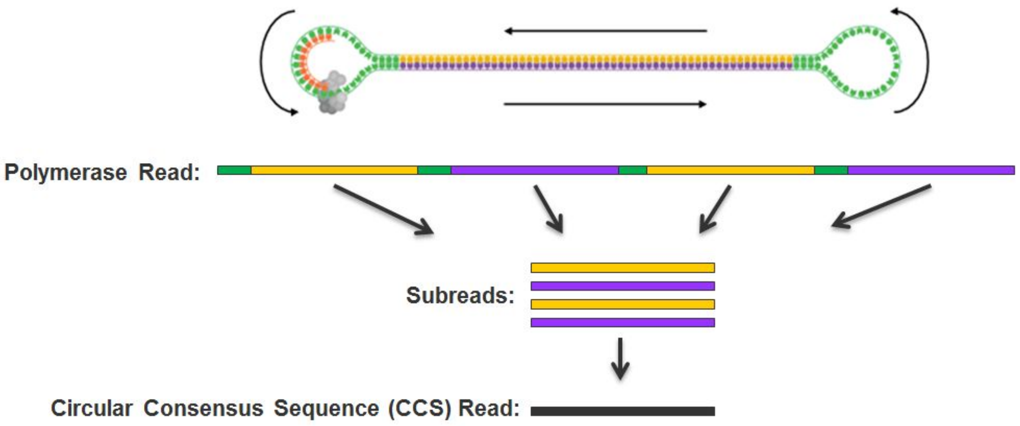
Long-read RNAseq from Pacific Biosciences (PacBio) and Oxford Nanopore Technologies (ONT) can sequence full-length transcripts, as well as large fragments.

PacBio Iso-Seq

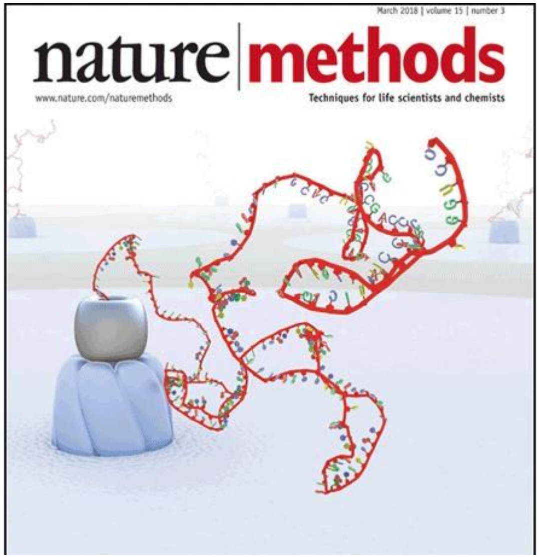
- Sequences cDNA multiple times to achieve higher-quality consensus read

ONT direct RNA-seq

- Sequences RNA molecule directly, enabling detection of modifications and structure



files.pacb.com/software/smrtanalysis/2.2.0/doc/smrtportal/help/!SSL!/Webhelp/Portal_PacBio_Glossary.htm

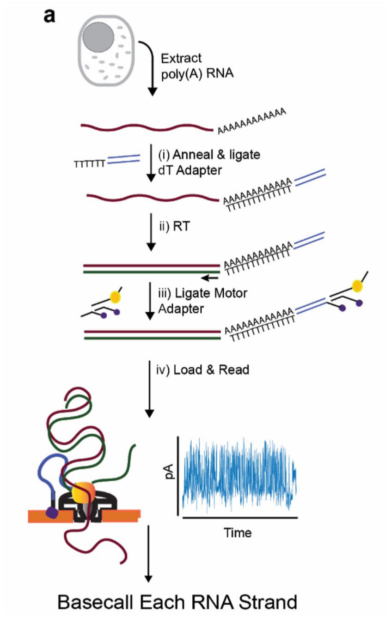

March 2018 | volume 15 | number 3
nature methods
www.nature.com/naturemethods
Techniques for life scientists and chemists

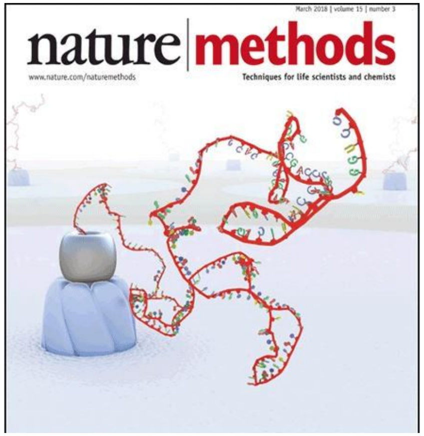
Nanopore Direct RNAseq

cDNA sequencing erases RNA modifications and secondary structure

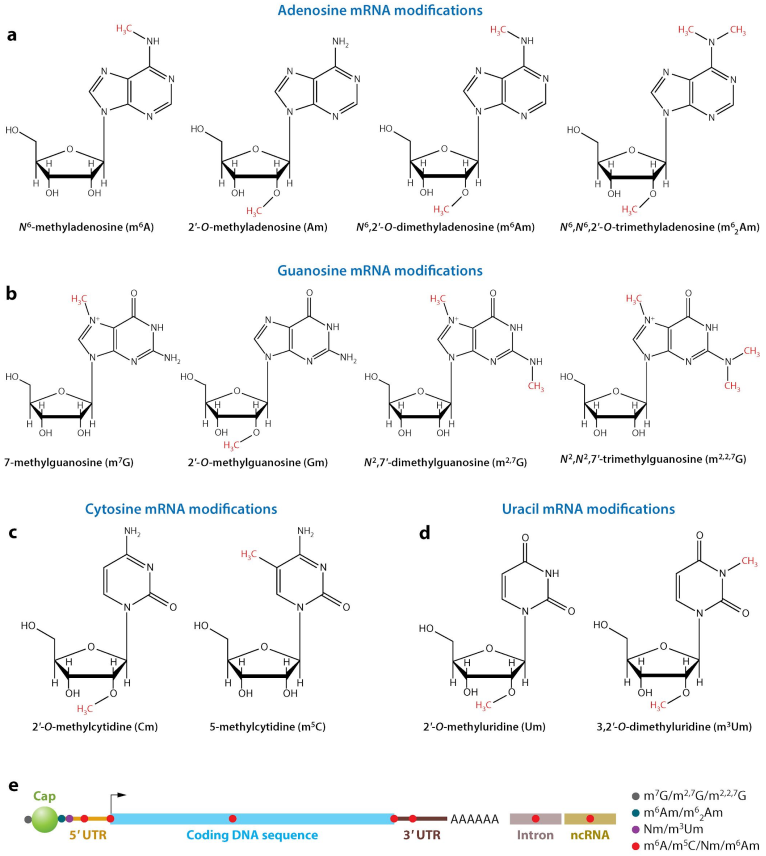
ONT direct RNAseq has potential to read both

On the other hand, secondary structure could also clog up the pore



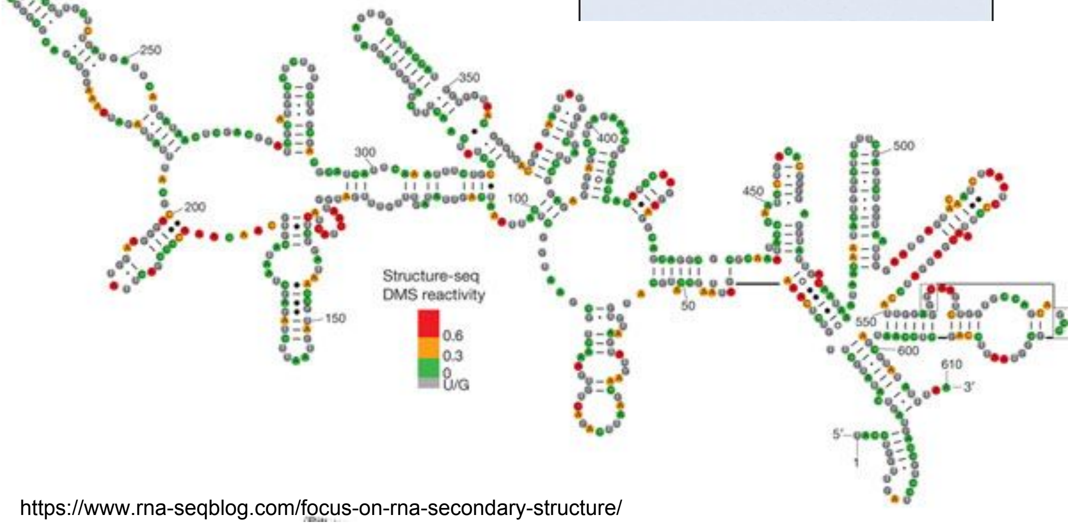


March 2018 | volume 15 | number 3
nature methods
www.nature.com/naturemethods
Techniques for life scientists and chemists



Adenosine mRNA modifications
a
N6-methyladenosine (m6A)
2′-O-methyladenosine (Am)
N6,2′-O-dimethyladenosine (m6Am)
N6,N6,2′-O-trimethyladenosine (m62Am)
Guanosine mRNA modifications
b
7-methylguanosine (m7G)
2′-O-methylguanosine (Gm)
N2,7′-dimethylguanosine (m2,7G)
N2,N2,7′-trimethylguanosine (m2,2,7G)
Cytosine mRNA modifications
c
2′-O-methylcytidine (Cm)
5-methylcytidine (m5C)
Uracil mRNA modifications
d
2′-O-methyluridine (Um)
3,2′-O-dimethyluridine (m3Um)
e
Cap
5′ UTR
Coding DNA sequence
3′ UTR
AAAAAA
Intron
ncRNA
m7G/m2,7G/m2,2,7G
m6Am/m62Am
Nm/m3Um
m6A/m5C/Nm/m6Am

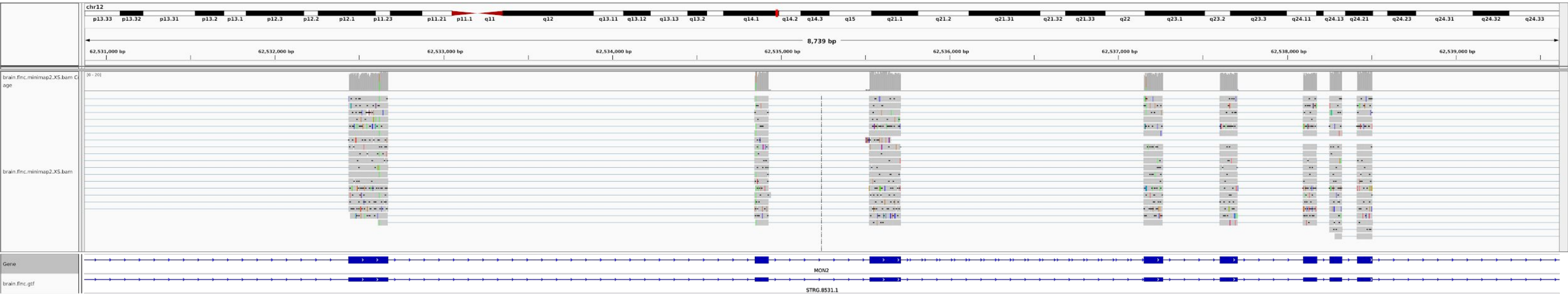
Li S, Mason CE. 2014.
Annu. Rev. Genomics Hum. Genet. 15:127–50



https://www.rna-seqblog.com/focus-on-rna-secondary-structure/

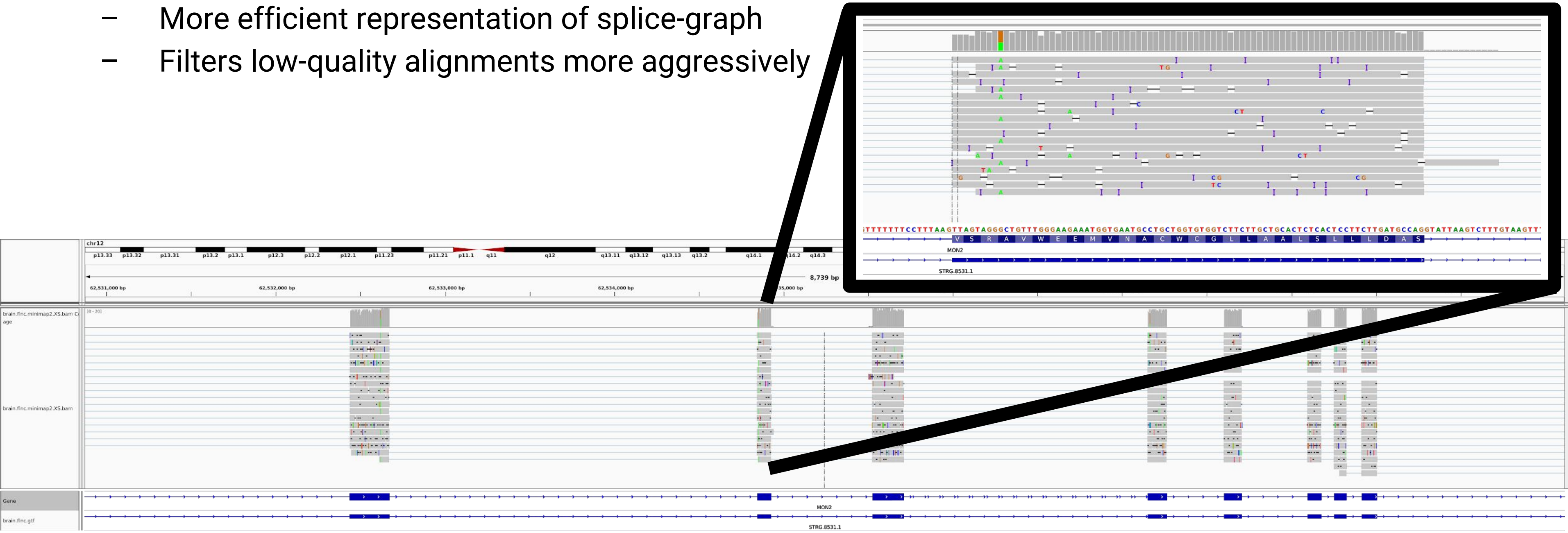
StringTie2

- Upcoming sequel to StringTie
- Outperforms Scallop on short reads
- Supports super-reads - “synthetic” long-reads
- Can assemble noisy PacBio and ONT reads
  - High frequency of indels makes splice graph more complicated, more spurious splice-sites
  - Corrects errors by forming consensus splice-sites
  - More efficient representation of splice-graph
  - Filters low-quality alignments more aggressively



StringTie2

- Upcoming sequel to StringTie
- Outperforms Scallop on short reads
- Supports super-reads - “synthetic” long-reads
- Can assemble noisy PacBio and ONT reads
  - High frequency of indels makes splice graph more complicated, more spurious splice-sites
  - Corrects errors by forming consensus splice-sites
  - More efficient representation of splice-graph
  - Filters low-quality alignments more aggressively



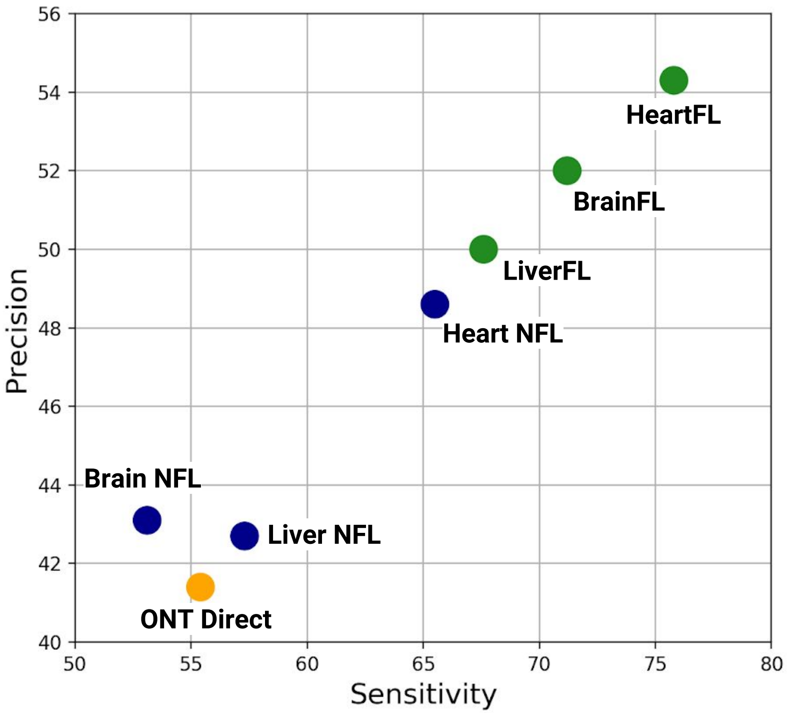
StringTie2 on Long Reads

Tested StringTie2 on seven human long-read datasets aligned using minimap2

- Three "full-length" PacBio datasets
- Three non-full-length PacBio datasets
- One ONT direct RNA-seq datasets (NA12878 consortium)

Example datasets provided by PacBio

To estimate sensitivity considered reference transcripts with ≥ 2x average coverage, ≥ 3x coverage surrounding introns



Welcome to Applied Comparative Genomics

https://github.com/schatzlab/appliedgenomics2019

Questions?