Sequence Analysis 2

B. Analysis of high-throughput sequences

Cheong Xin Chan (CX)

c.chan1@uq.edu.au

Australian Centre for Ecogenomics
School of Chemistry & Molecular Biosciences
The University of Queensland

SCIE2100 | BINF6000 | Bioinformatics 1 – Introduction

Lecture outline

- **High-throughput sequence data**
  - The data (sequence reads and quality)
  - Types of assemblies
- **Basic principles of genome assembly**
- **Strategies of genome assembly**
  - Overlap-layout-consensus
  - De Bruijn graph (*k*-mer-based, and examples)
  - Key terms and concepts
- **Issues and challenges**

High-throughput sequences

High-throughput sequences are short. Why?

Current sequencing technologies are not practical enough to read whole genomes in one go.

Shotgun sequencing breaks down genome into small fragments (e.g. 10^2 bases) then sequence these fragments in great depth (typically 10^6-10^8 sequence reads; hence **high throughput**).

These reads will need to be assembled to *re-assemble* the original genome sequence.

How?

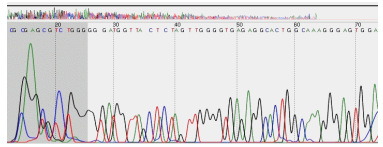
Identical regions of different reads can be collapsed into a long contiguous sequence (*à la* sequence alignment)

Regions of bad reads (with low quality score) can be down-weighted

How do we assess data quality?

A (very) simplified sequencing workflow

Sequencing



Trace data

Base calling

Sequence reads

Phred quality score (*Q*)

$$Q = -10(\log_{10}P)$$

where *P* = probability of a base-calling error

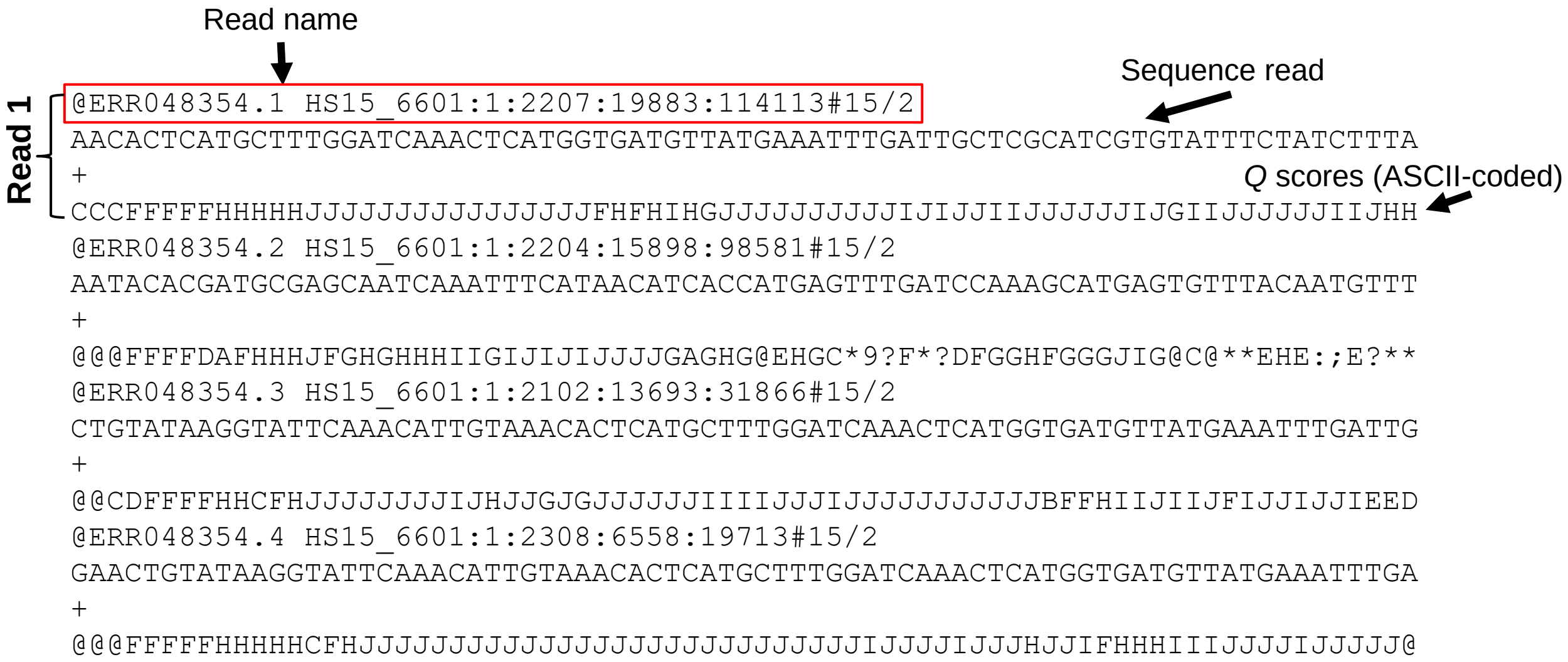
Phred Quality Score	Probability of Incorrect Base Call	Base Call Accuracy
10	1 in 10	90%
20	1 in 100	99%
30	1 in 1,000	99.9%
40	1 in 10,000	99.99%
50	1 in 100,000	99.999%

http://www.illumina.com/documents/products/technotes/technote_Q-Scores.pdf

$P = 0.001$

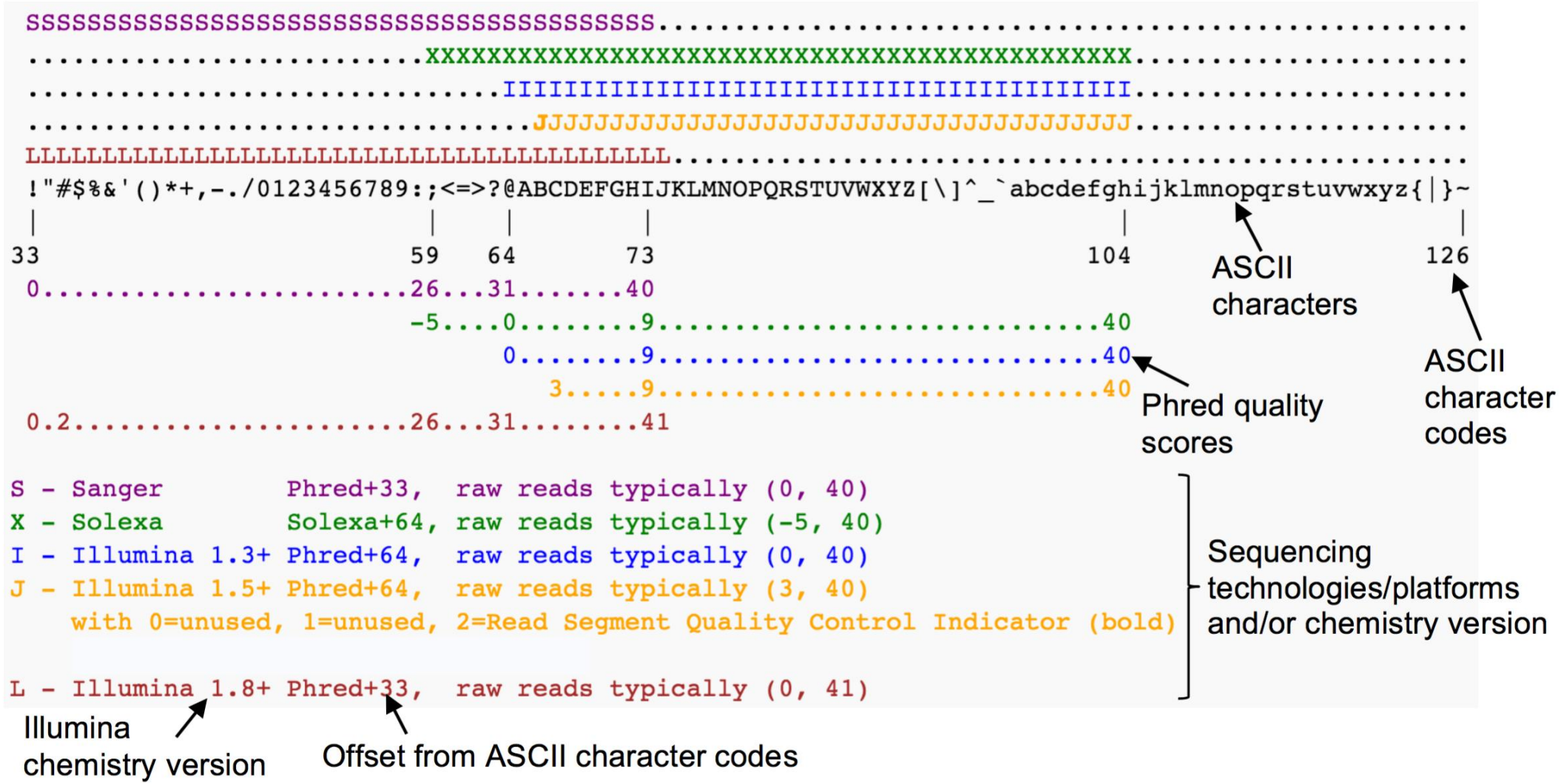
$Q = -10(\log_{10}[0.001]) = -10(-3) = 30$

High-throughput data: sequence reads



FASTQ format

Q scores in ASCII characters



Why is the offset 33?

The first 32 ASCII characters are non-printing characters (*e.g. esc, tab, backspace, ctrl, shift etc.*)

How can we tell if our data are good?

```
@ERR048354.1 HS15_6601:1:2207:19883:114113#15/2
AACACTCATGCTTTGGATCAAACTCATGGTGATGTTATGAAATTTGATTGCTCGCATCGTGTATTTCTATCTTTA
+
CCCFFFFFHHHHHJJJJJJJJJJJJJJJJJFHFHIHGJJJJJJJJJJJIJIJJIIJJJJJJIJGIIJJJJJIIJHH
@ERR048354.2 HS15_6601:1:2204:15898:98581#15/2
AATACACGATGCGAGCAATCAAATTTCATAACATCACCATGAGTTTGATCCAAAGCATGAGTGTTTACAATGTTT
+
@@@FFFFDAFHHHJFGHGHHHIIGIJIJIJJJJGAGHG@EHGC*9?F*?DFGGHFGGGJIG@C@**EHE:;E?**
@ERR048354.3 HS15_6601:1:2102:13693:31866#15/2
CTGTATAAGGTATTCAAACATTGTAAACACTCATGCTTTGGATCAAACTCATGGTGATGTTATGAAATTTGATTG
+
@@CDFFFFHHCFHJJJJJJJJIJHJJGJGJJJJJJIIIIJJJIJJJJJJJJJJJJBFFHIIJIIJFIJJIJJIEED
@ERR048354.4 HS15_6601:1:2308:6558:19713#15/2
GAACTGTATAAGGTATTCAAACATTGTAAACACTCATGCTTTGGATCAAACTCATGGTGATGTTATGAAATTTGA
+
@@@FFFFFHHHHHCFHJJJJJJJJJJJJJJJJJJJJJJJJJJJJJIJJJJIJJJHJJIFHHHIIIJJJJIJJJJJ@
...
```

Technical: base quality of the reads, presence of unwanted adapters or artefacts, sequencing errors/biases, etc.

Biological: presence of unwanted/contaminant sequence reads, adequacy of the data in addressing biological questions, etc.

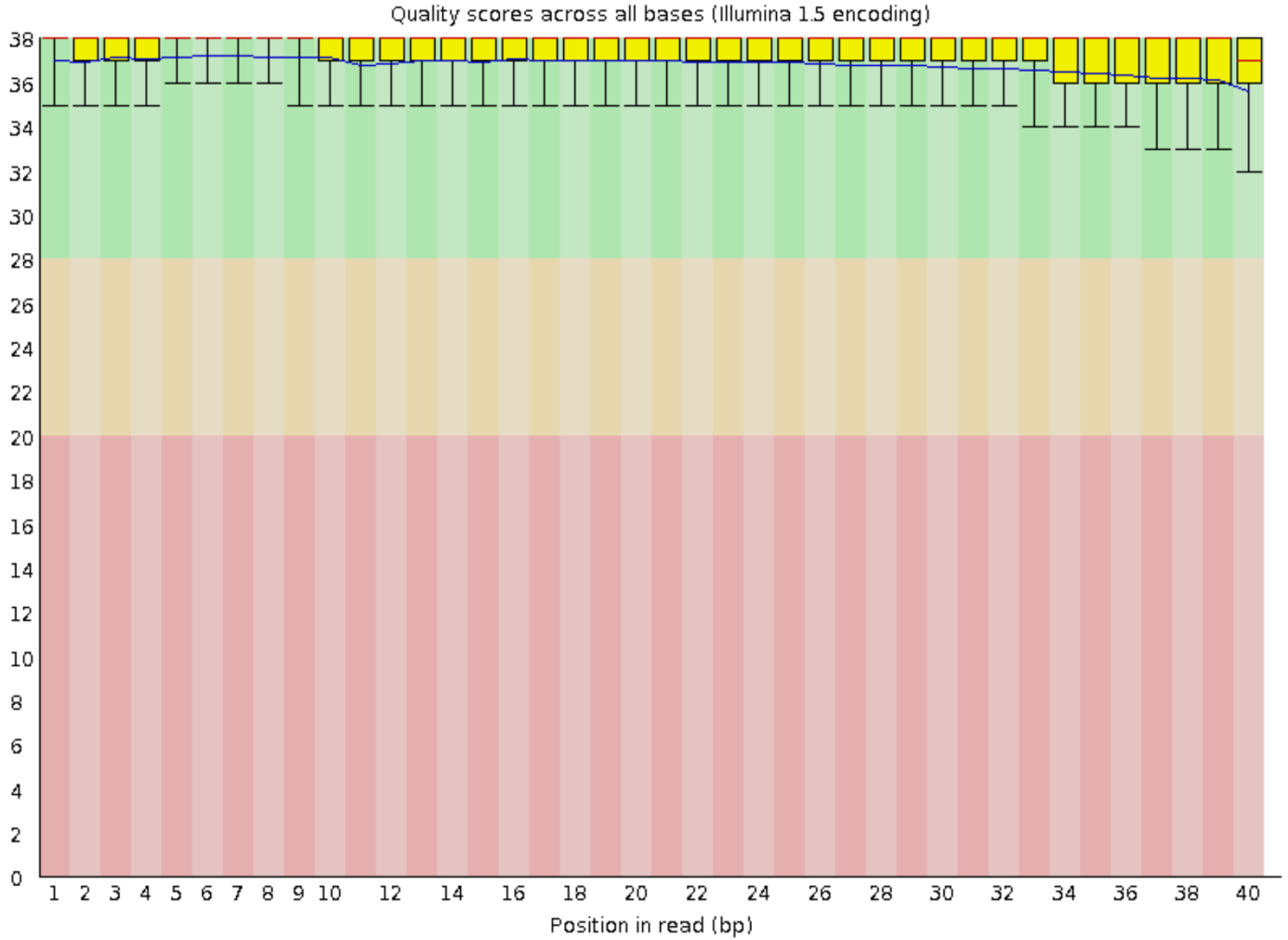
In an ideal world …

FastQC Report

Summary

- Basic Statistics
- Per base sequence quality
- Per tile sequence quality
- Per sequence quality scores
- Per base sequence content
- Per sequence GC content
- Per base N content
- Sequence Length Distribution
- Sequence Duplication Levels
- Overrepresented sequences
- Adapter Content

Per base sequence quality



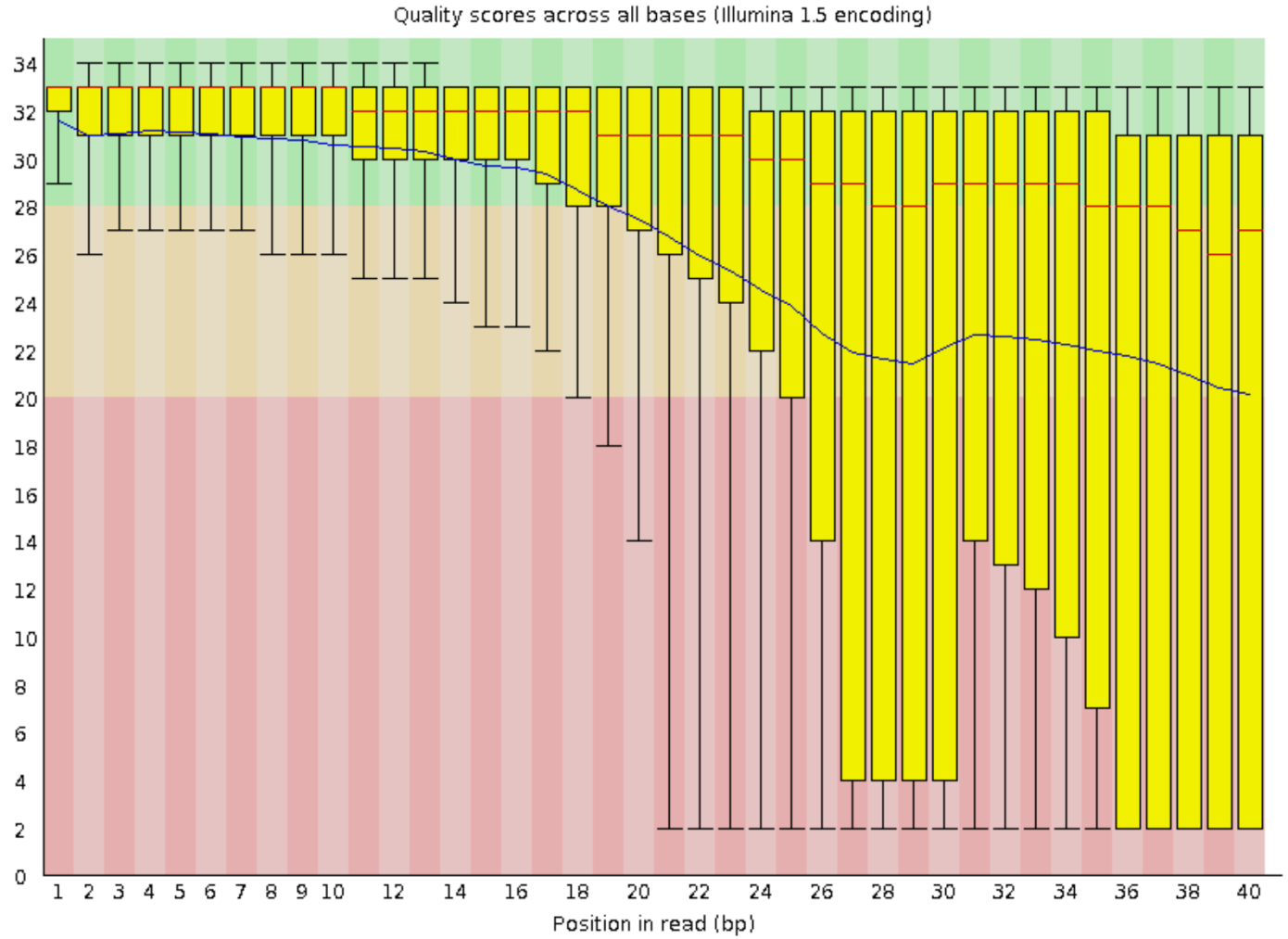
In reality …

FastQC Report

Summary

- Basic Statistics
- Per base sequence quality
- Per tile sequence quality
- Per sequence quality scores
- Per base sequence content
- Per sequence GC content
- Per base N content
- Sequence Length Distribution
- Sequence Duplication Levels
- Overrepresented sequences
- Adapter Content

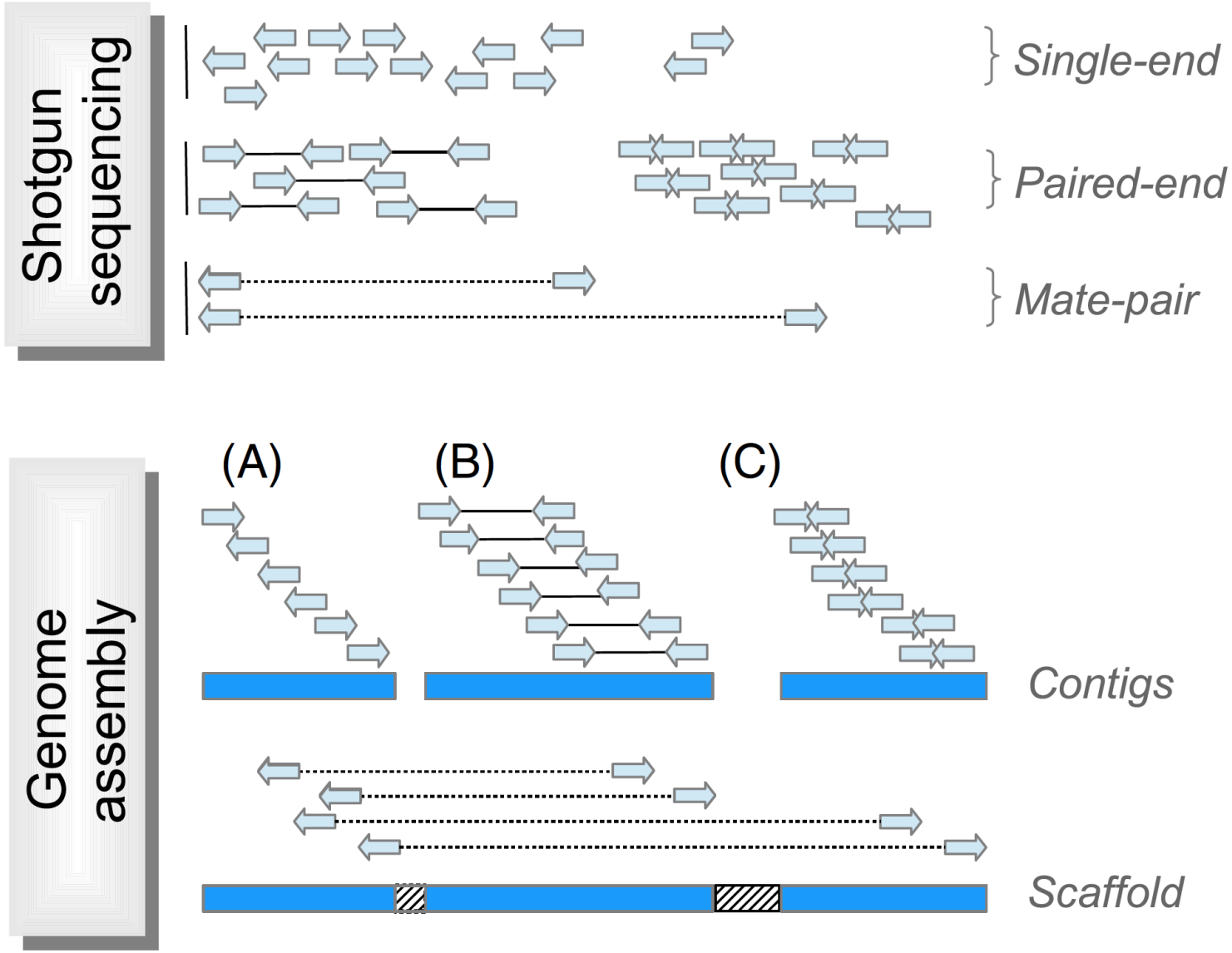
Per base sequence quality



Overrepresented sequences

Sequence	Count	Percentage	Possible Source
CGGTTCAGCAGGAATGCCGAGA TCGGAAGAGCGGTTCAGC	12351	0.5224039	Illumina Paired End PCR Primer 2 (96% over 25bp)

Basic principles of genome assembly



Ekblom & Wolf (2014) *Evolutionary Applications* 7: 1026-42.

Genome *versus* transcriptome assemblies

Genome assembly

- Genome include genic and intergenic regions
- Often have large amounts of repetitive elements, esp. in the intergenic regions (e.g., introns)
- Long contigs are desirable

CLC Genomics Workbench
MaSuRCA
ALLPATHS-LG
SPAdes
Velvet
Celera
Newbler (for 454 data)
ABySS
SOAPdenovo
…

Transcriptome assembly

- Expressed genes - no intergenic regions
- No/less repetitive elements
- Housekeeping genes are highly expressed (higher coverage/more reads)
- Complication of alternative splicing, SNP, and post-transcriptional modification
- Long contigs might indicate multigene cluster due to over-assembly

Trinity
Velvet-Oases
Newbler (-cdna option)
CAP3
…

De novo versus mapping assemblies

De novo assembly

- Assembling a new, previously unknown sequence or genome
- More memory intensive due to computational complexity

New genomes

Mapping assembly

- Assembling reads against a reference sequence/genome
- Looking for an assembly that is similar (not necessarily identical) to the reference

- Resequencing projects
- Genomes of similar species
- Model genomes of animals/plants (e.g., humans, *Arabidopsis*)
- 1000-genome projects

…

Strategy of genome assembly

The **shortest common superstring** problem

Given a collection of strings, what is the shortest superstring that contains all these strings as substrings?

Example: *S*: BAA AAB BBA ABA ABB BBB AAA BAB

Concatenation: BAAAABBBAABAABBBBBAAABAB

24

Shortest common superstring: AAABBBABAA

10

AAA
AAB
ABB
BBB
BBA
BAB
ABA
BAA

*Given a collection of **sequence reads**, what is the **shortest sequence** that contains **all these reads** (as sub-sequences)?*

Strategy of genome assembly

Two major paradigms:

Overlap-layout-consensus (OLC)

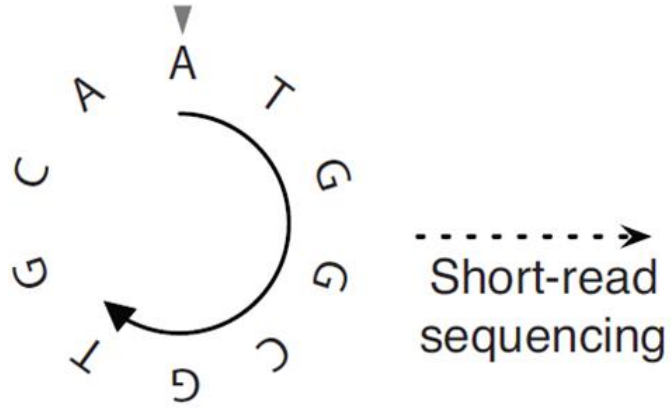
- identifies all pairs of reads that overlap sufficiently well and then organises this information into a graph
- e.g. Celera

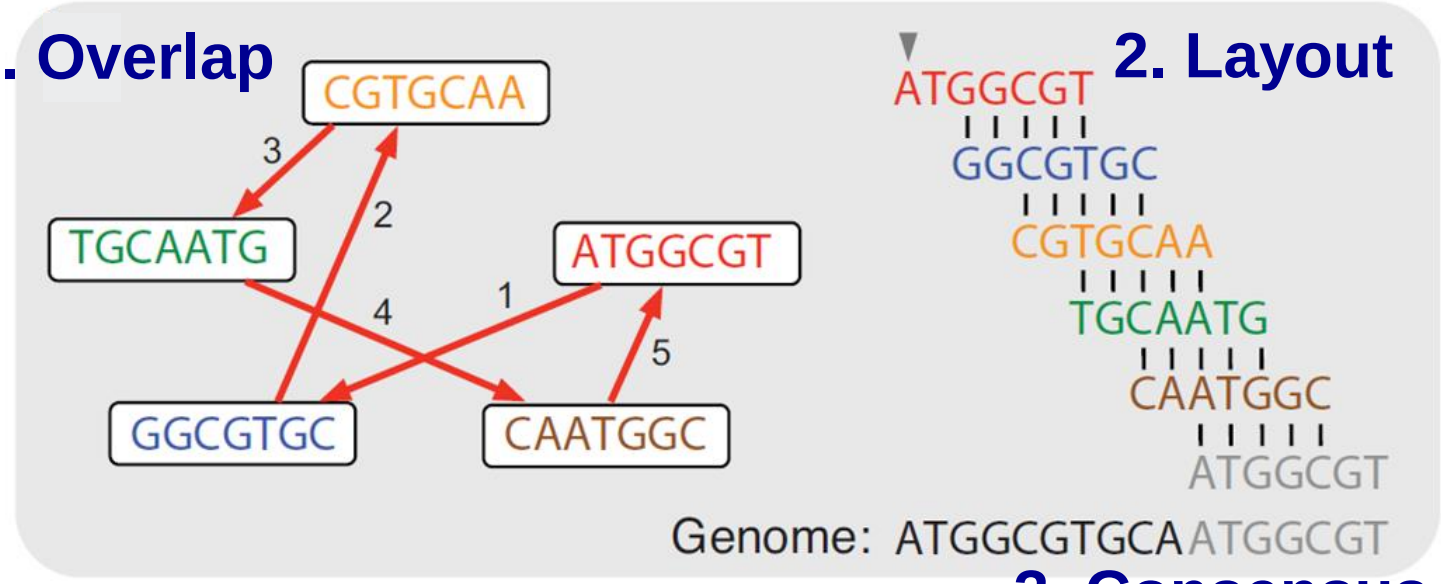
De Bruijn graph

- models the relationship (overlaps) between exact substrings of length *k* extracted from the input reads (*k*-mers)
- More popular; e.g. CLC-GW, MaSuRCA, ALLPATHS, SPAdes, Velvet, SOAPdenovo

Nagarajan & Pop (2013) *Nature Reviews Genetics* 14: 157-167.

Overlap-layout-consensus (OLC)





- Reads represented as **nodes**; alignments between reads as **edges**
- Genome reconstruction by combining alignments between successive reads
- Computationally expensive

Compeau *et al.* (2011) *Nature Biotechnology*, 29: 987-991.

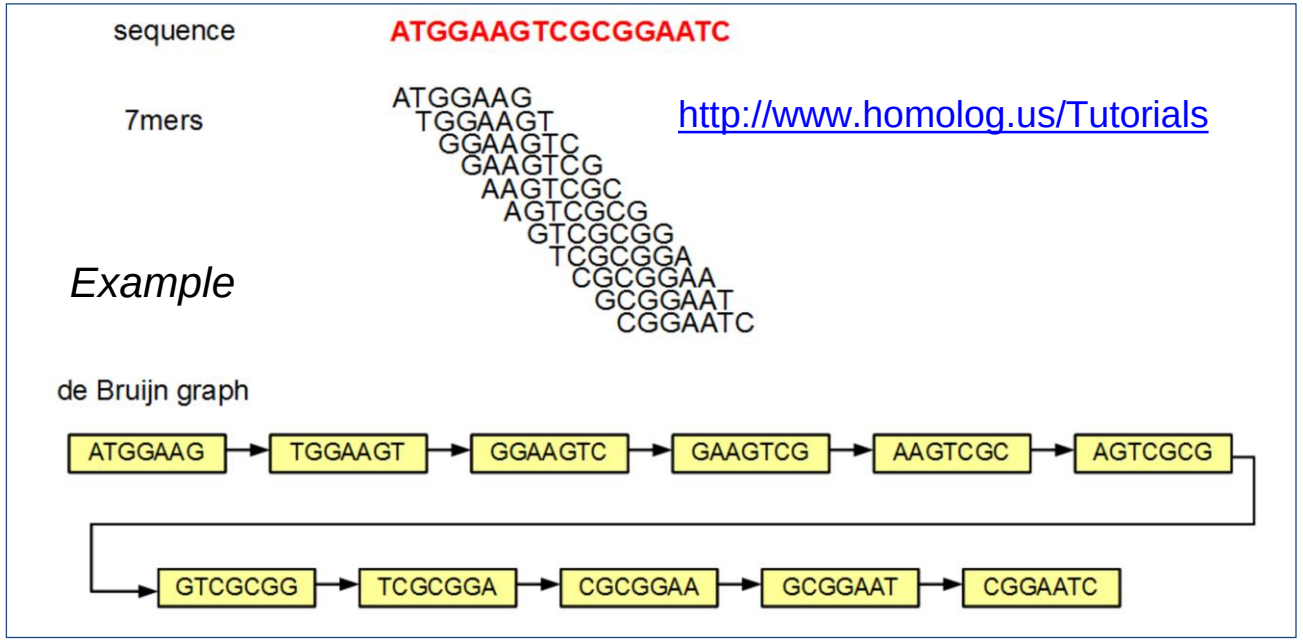
De Bruijn graph (based on *k*-mers)

What are *k*-mers?

TTGACACTTACCGA **Read**

TTGACACTTACC
TGACACTTACCG
GACACTTACCGA
k-mers for k=12

TTGAC
TGACA
GACAC
ACACT
CACTT
ACTTA
CTTAC
TTACC
TACCG
ACCGA
k-mers for k=5



- Obtain all *k*-mers at (*k* = 7 in this case) – these are **nodes**
- Construct directed graph between node pairs such that the connected nodes have (contiguous) overlaps of 6 (*k* - 1) nucleotides; these connections are **edges**.
- Find the **shortest (common) superstring**

De Bruijn graph

In a very simple example …

Example #1:

HAPPI PINE INESS APPIN

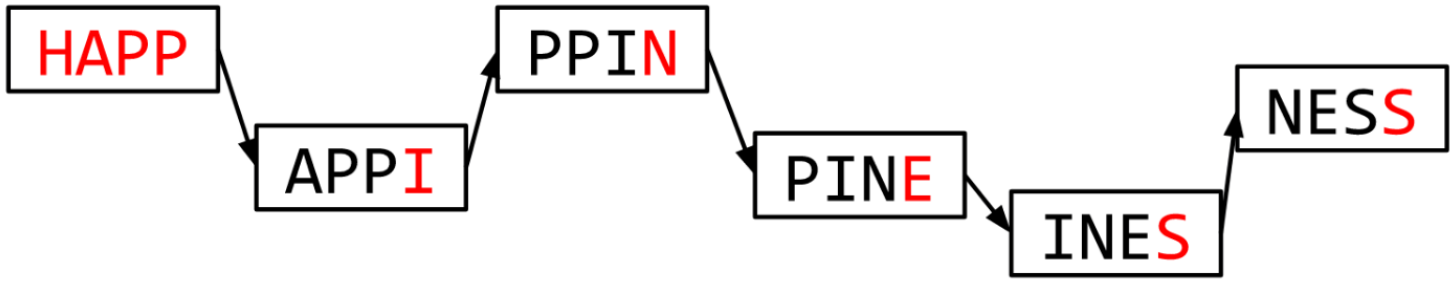
All 4-mers:

HAPP PINE INES APPI

APPI NESS PPIN

Unique 4-mers:

HAPP APPI PINE PPIN INES NESS



HAPPINESS

Identical **nodes** are merged/collapsed, reducing computational complexity.

De Bruijn graph

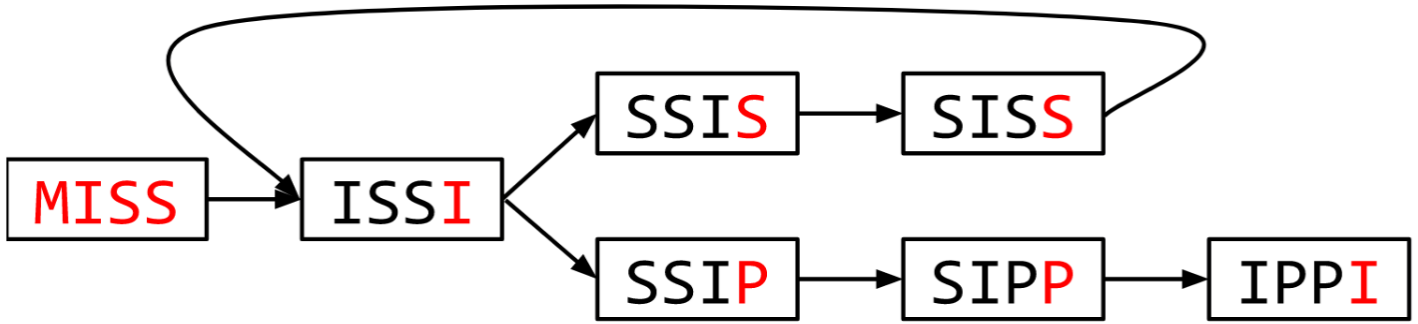
In a more-tricky example when repetitive sequence regions are present …

Example #2:

`MISSIS SSISSI SSIPPI`

All 4-mers (9):

```
MISS   SSIS   SSIP
 ISSI   SISS   SIPP
  SSIS   ISSI   IPPI
```



`MISSISSIPPI` or `MISSISSISSISSIPPI` or ...

Unique 4-mers (7):

`MISS SSIS SSIP ISSI SISS SIPP IPPI`

The same **node** can be used in assembling different sequences.

De Bruijn graph

Example #2a:

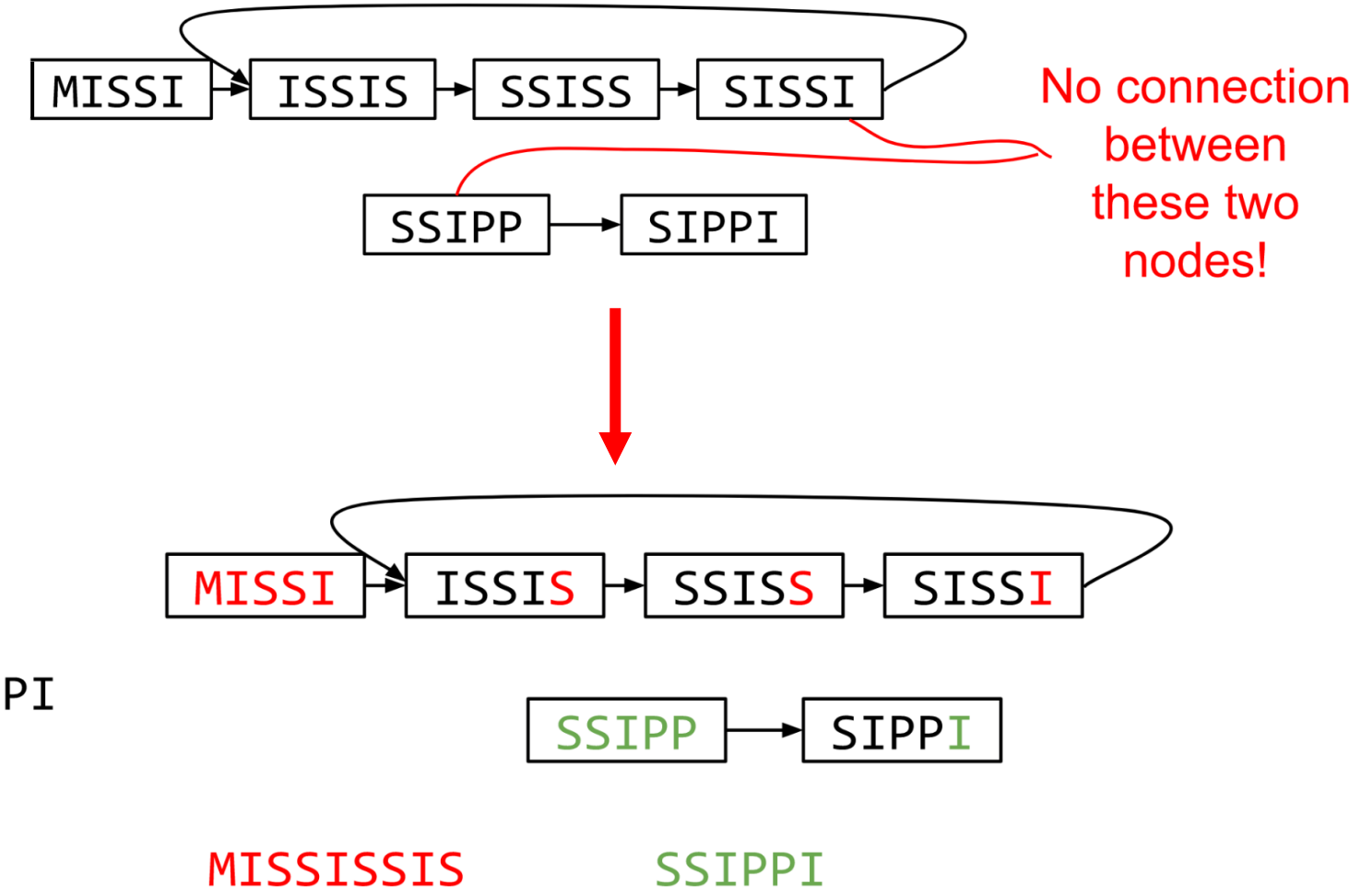
```
MISSIS SSISSI SSIPPI
```

All 5-mers (6):

```
MISSI  SSISS  SSIPP
 ISSIS  SISSI  SIPPI
```

Unique 5-mers (6, no duplicates):

```
MISSI ISSIS SSISS SISSI SSIPP SIPPI
```



Different *k* values yield different results.

k-mer length in de Bruijn graph

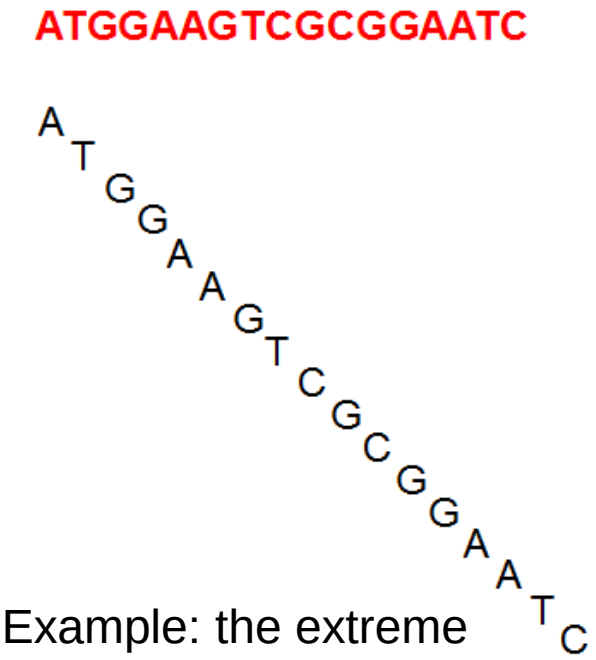
The choice of *k*-mer length (*k*) is crucial:

Short *k*:

- lowers overlap threshold (more overlaps), *k*-mers joined more readily
- generates large number of short contigs

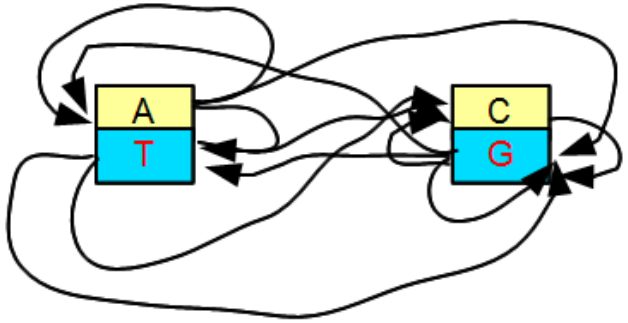
Long *k*:

- may not join contigs together when it should



Example: the extreme case of using 1-mers

It is difficult to predict the best *k* to use for each assembly, thus sometimes optimisation by trial-and-errors is necessary.



k-mer length in de Bruijn graph

7-mer (**ATATATA**):

-> ATATATA

TATATAT <-

Reverse complement of **ATATATA** is **TATATAT**

okay

6-mer (**ATATAT**):

-> ATATAT

TATATA <-

Reverse complement of **ATATAT** is **ATATAT**

confusing

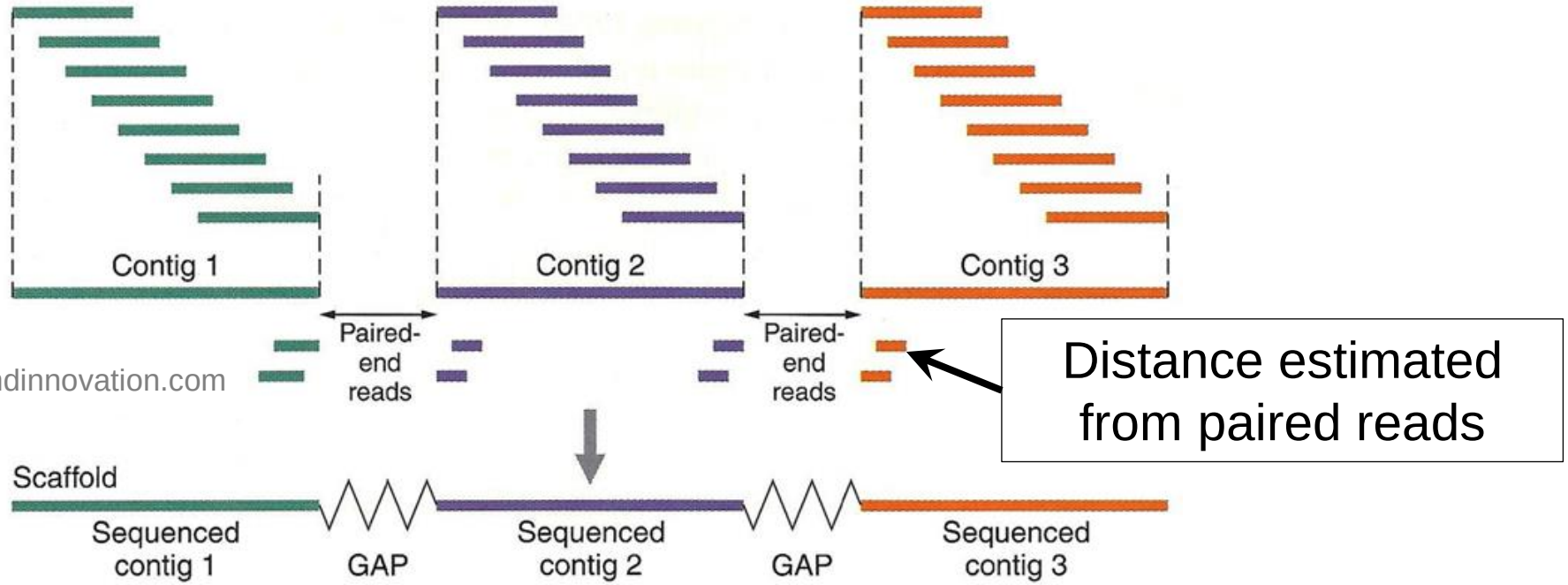
k is commonly an odd number to avoid palindromes

- If *k*-mers are of even length, some *k*-mers can be reverse complements of themselves (e.g. **ATATAT**)
- Genome assemblers commonly avoid even *k*

k must be shorter than the sequence read length, or else there will be no overlaps

Key terms and concepts

Contig: a contiguous linear stretch of consensus sequence that is constructed from a number of smaller, partially overlapping, sequence reads (fragments)



Scaffold: a sequence comprising two or more contigs that are joined together based on read-pair distance information (i.e. ordered, oriented contigs with NNNs in between)

Coverage (sequencing depth): the average number of reads representing a given nucleotide in an assembled sequence (e.g. genome)

Example: a contig of 30 bases

```
Read1       GATCTGGAATTCTCGGGCAC
Read2          CTGGAATTCTCGGGCACCAA
Read3           TGGAATTCTCGGGCACCAAG
Read4                 TCTCGGGCACCAAGGTACGC
Contig      GATCTGGAATTCTCGGGCACCAAGGTACGC
Base        11123333334444444443332111111
Coverage
```

4 reads of 20 bases = 80 bases constitute this 30-base contig

Contig coverage = **80 / 30** = **2.67**

N50: Contig length such that using equal or longer contigs produces half (50%) of the bases of the total assembled bases (sum of all contig lengths); the same applies to scaffolds

Example: a genome assembly

Contig	*Length*	*Cumulative Sum*
Contig3	295,492	295,492
Contig2	259,553	555,045
Contig6	142,866	697,911
Contig1	136,171	834,082
Contig9	135,129	969,211
Contig7	117,473	1,086,684
Contig10	115,625	1,202,309
Contig4	102,105	1,304,414
Contig8	77,713	1,382,127
Contig5	76,819	**1,458,946**

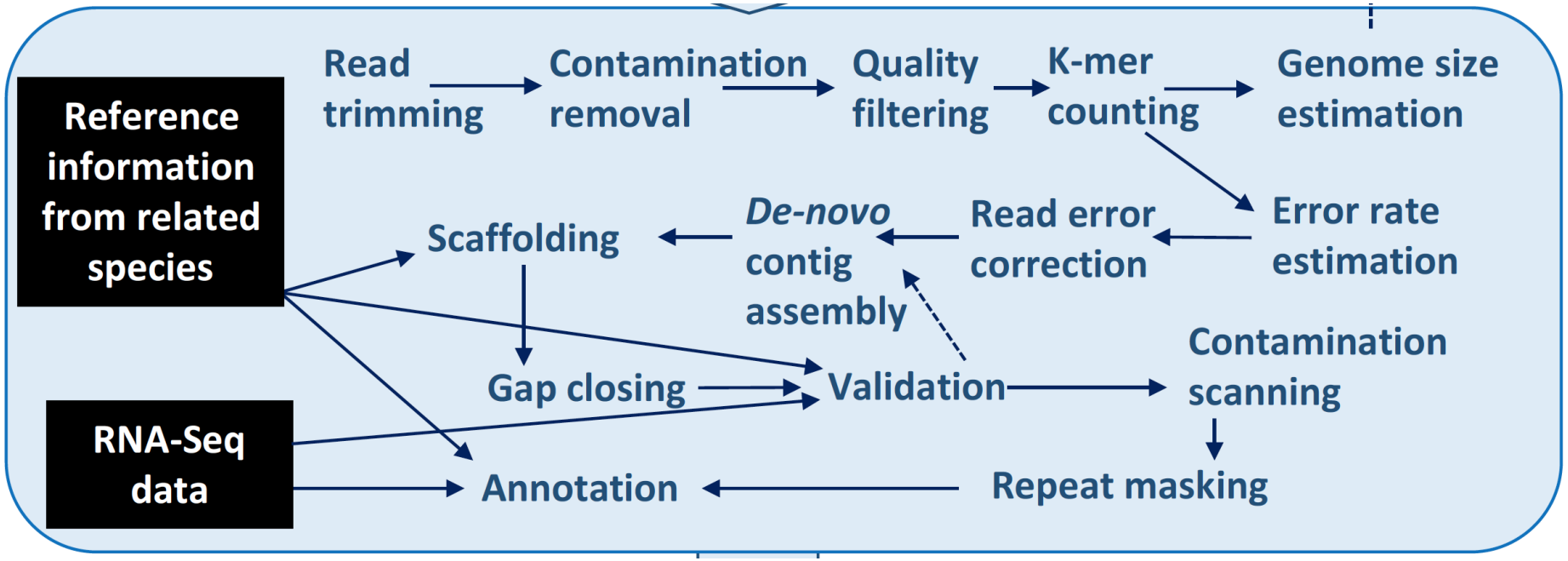
50% of bases = 1,458,946 / 2 **= 729,473**

The **first four** contigs make up ≥ 729473 bases therefore **N50 = 136,171**

How about N75?

25% of bases = 1,458,946 / 4 **= 364,736.50**

The first **two** contigs make up ≥ 364,736.50 bases. **N25** = **259,553**



A typical process of *de novo* genome assembly

Ekblom & Wolf (2014) *Evolutionary Applications* 7: 1026-42.

Read trimming: the removal of adapter sequences and low-quality/ambiguous bases from the sequence reads

Read mapping: the alignment of short sequence reads to a longer sequence (e.g. contig, scaffold, reference genome)

Gap filling/closing: the process of replacing the Ns in scaffolds with nucleotide bases based on read mapping

Issues and challenges

- Peculiarity (little-understood features) in genomes, prokaryotes versus eukaryotes
- Repetitive elements in genomes (cause error in assembly, increase time/space complexity)
- Computational tractability, memory and storage space, esp. when dealing with huge amount of data
- Sequencing error (e.g., assignment of incorrect base, under-/over-estimation of base quality scores
- All these intensified with *de novo* assemblies

Reflection

- *What are high-throughput sequence data?*
- *How do we assess the quality of high-throughput sequence data?*
- *What are the basic principles of genome assembly? Why do we need to assemble a genome (or transcriptome)?*
- *What are the differences between a genome assembly and a transcriptome assembly?*
- *What are the differences between a de novo assembly and a mapping assembly?*
- *What are the two major paradigms of genome assembly?*
- *What is a De Bruijn graph assembly?*
- *How do we choose the k-mer length in De Bruijn graph assembly?*
- *What are the key properties in genome assembly? How do we calculate N50 length?*
- *What are the key issues and challenges facing genome assembly?*