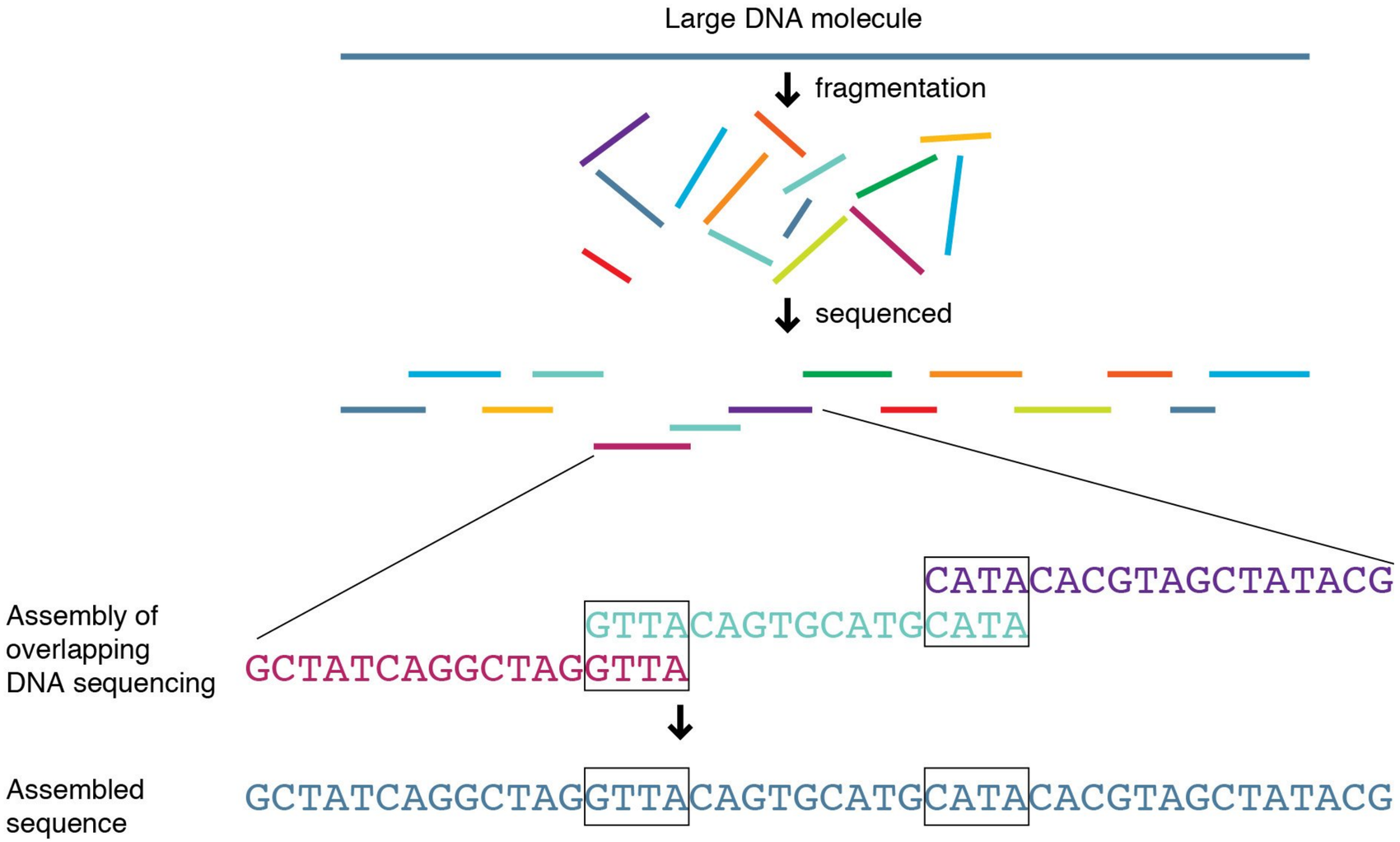
NIH Fogarty International Center
Advancing Science for Global Health

Sequence Assembly

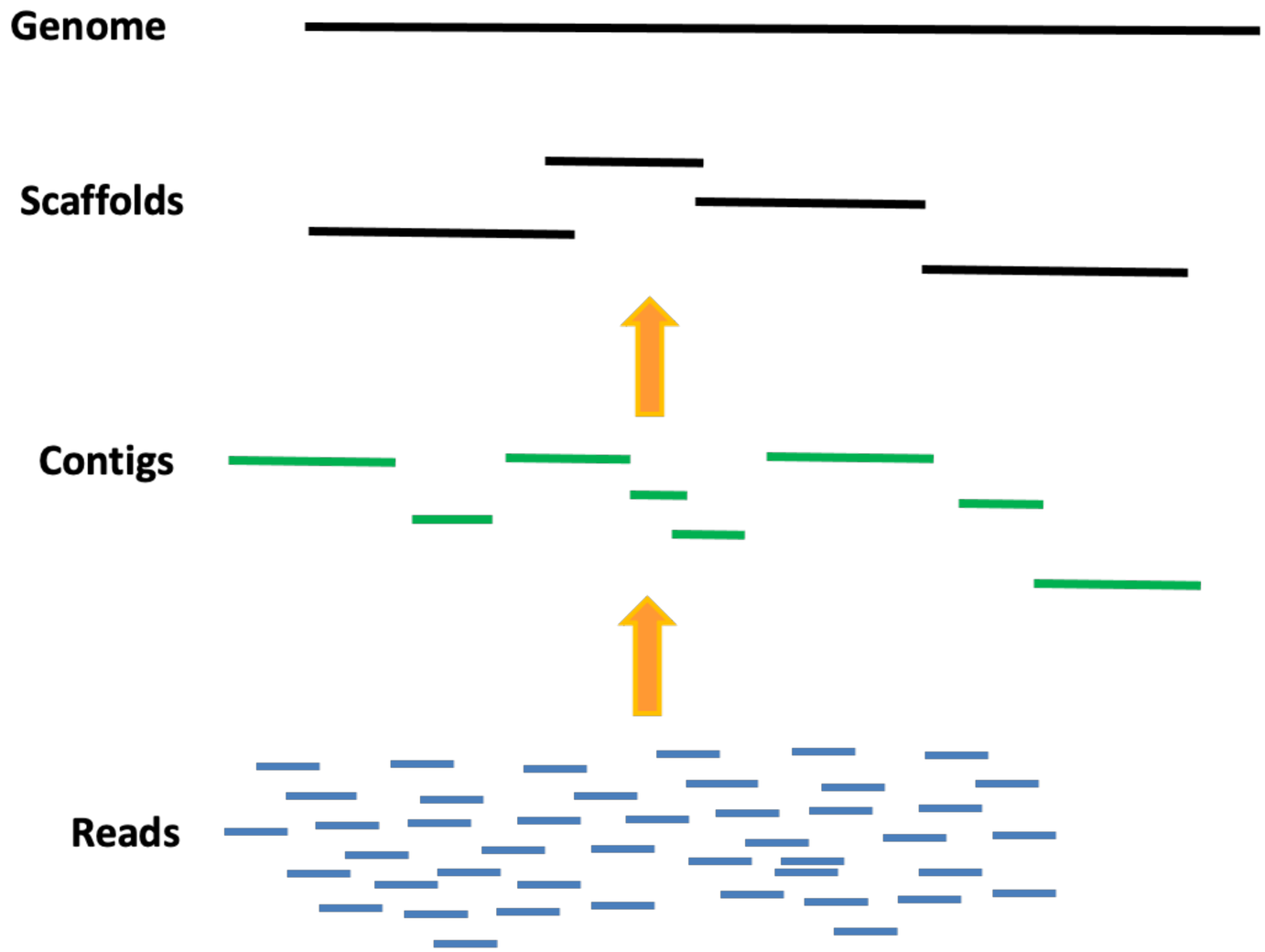
James Richard Otieno (PhD)
Division of International Epidemiology and Population Studies
Fogarty International Center
National Institutes of Health
Bethesda, Maryland, USA

NIH Fogarty

Genome Sequencing and Assembly



Assembly Overview



What is a sequence assembly?

An assembly is an hierarchical data structure that maps the sequence data to a ***putative*** reconstruction of the target

Miller JR, Koren S and Sutton G. 2010. Assembly algorithms for next-generation sequencing data. Genomics 95:315-327

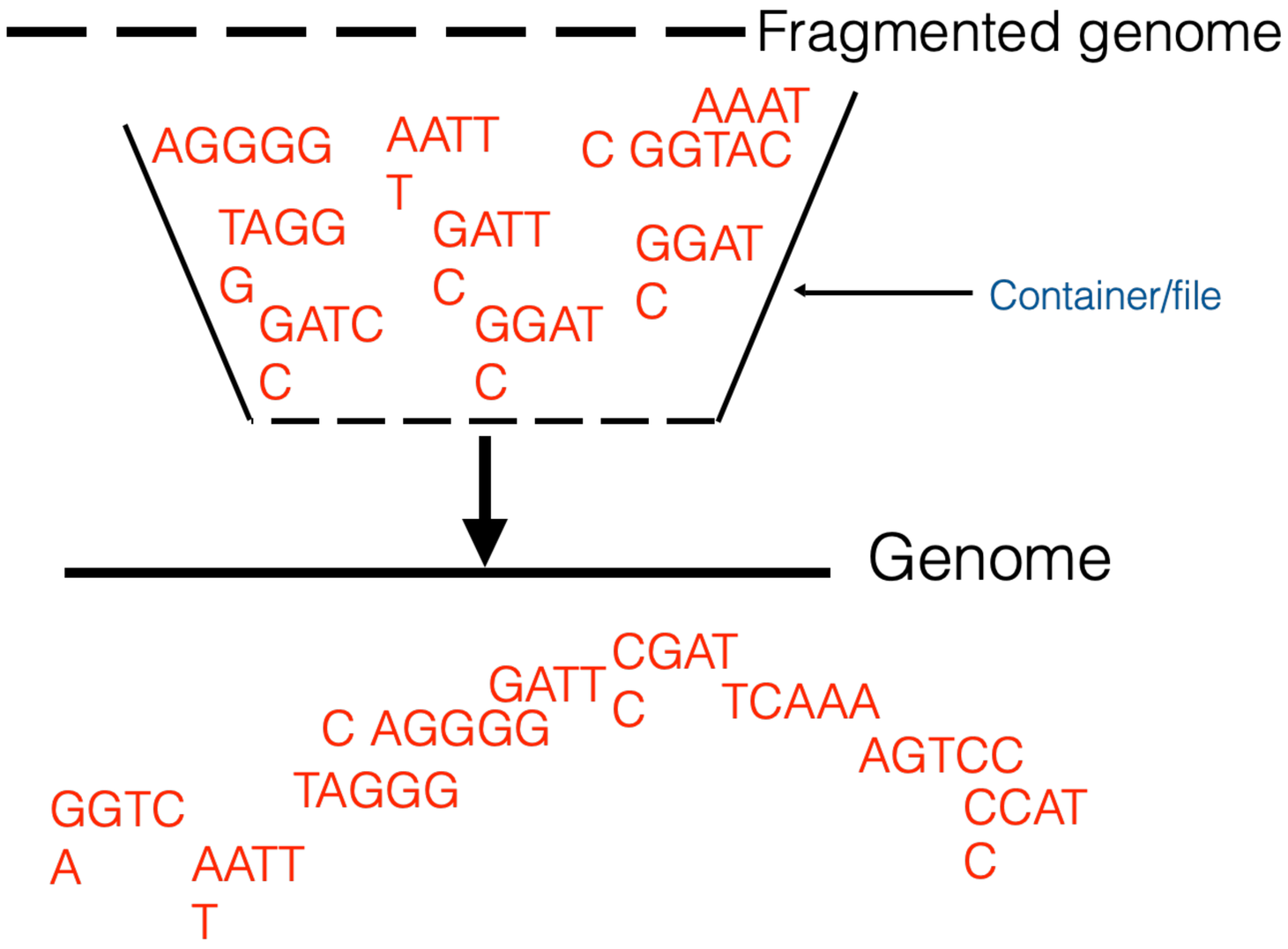
1. “Classical” sequence assembly

2. NGS sequence assembly strategies

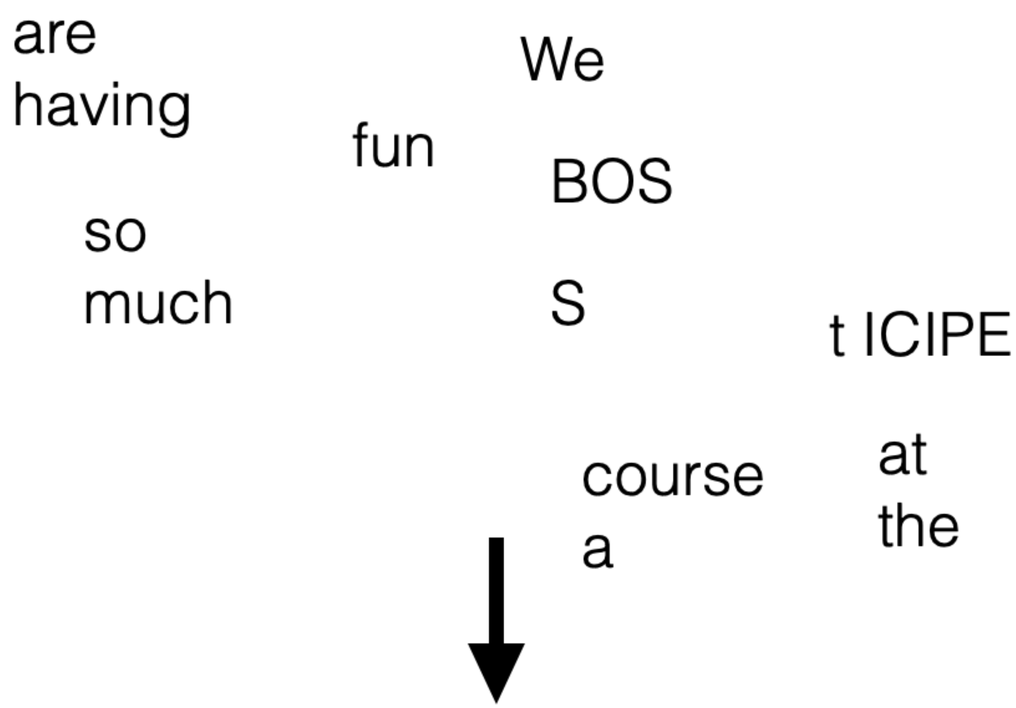
Data format

1. Remember that you fragmented your genome
2. So you end up with fragmented sequences of the genome (i.e. your NGS data)
3. We shall only consider data in **fastq** format
4. However, you may have data in a different format (fast5, SAM, BAM, etc)
5. It is possible to convert between formats

How your data looks



Another example



“We are having so much fun at the BOSS course at ICIPE”

This is the principle of assembly

How your data file actually looks

```
@read1
AGCTTATCCTCTGCTCACCCCCGGGTTAGCGCACTTGATGTATTCACAGC
+
BA1@CC7CBCCC9C8;B2@>C?B@B@B3=9?@B1:AB7B?B8B?B6B.7.
@read2
TTGGGCGGGATCTCCAGAAGCATATGGATGTGATCCACACAGCATTCTGC
+
?>?B@)<?@,AA7A@C<C?=@@B;+)?B5*@2=@+=BB,=B6C>AB@B24
@read3
TATGCTCAAGAAGGGGCTGATGAGTTGGTGTTTTACGATATCACTGCCTC
+
A3AB:B1:B;9/0BBBCBB<BB@AA0?BB9:BB<A@BB@7@6@<A@@@<3
```

“Classical” Sequence Assembly

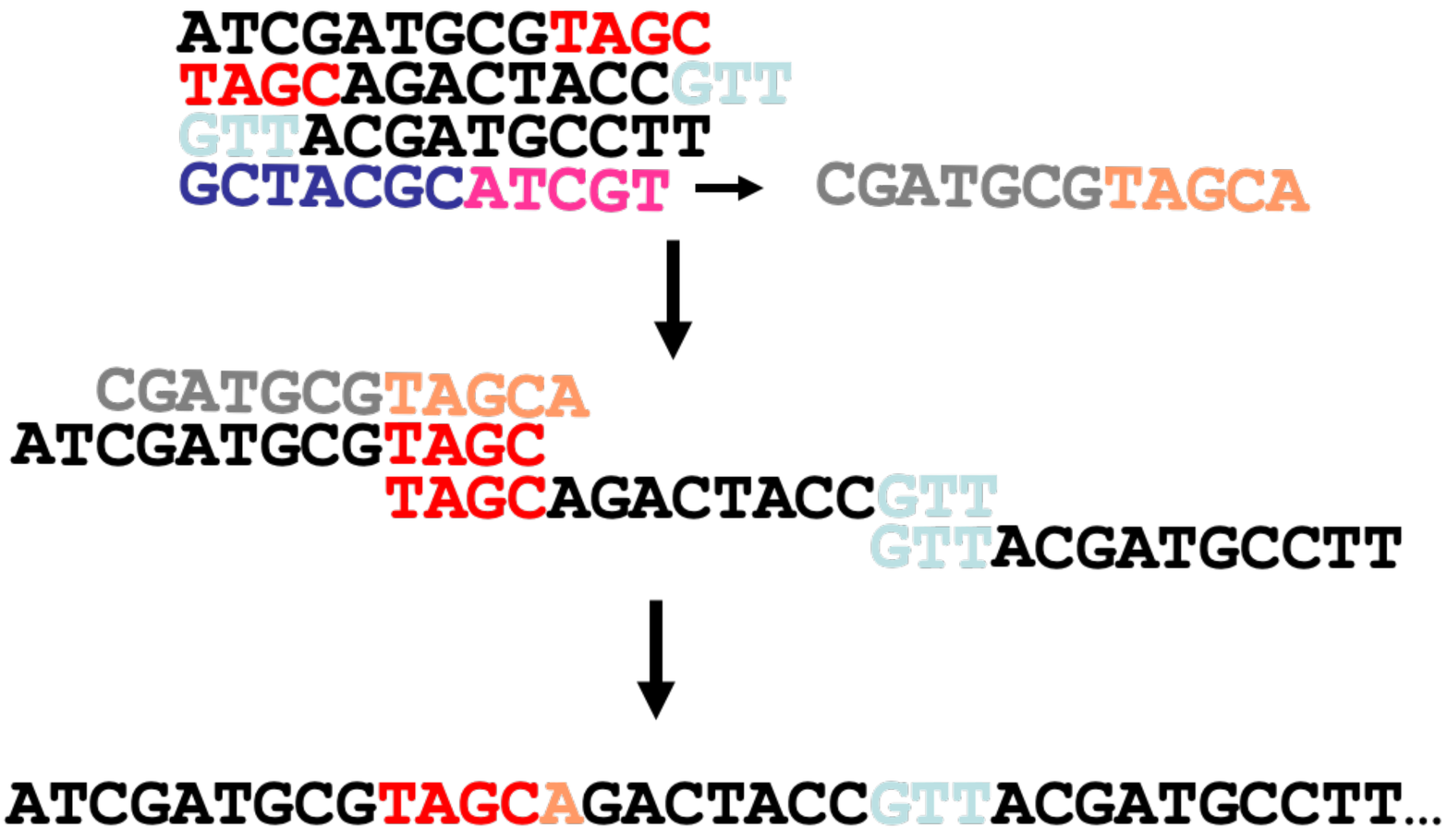
1. Read in all sequence files (10-10,000)
2. Read, edit & trim DNA chromatograms

   ~Remove vector sequences (vector trim)

   ~Remove regions of low complexity

   ~Remove overlaps & ambiguous calls
3. Perform multiple sequence alignment & merge
4. Fill (“finish”) gaps using a variety of experimental procedures.

“Classical” Sequence Assembly



NGS Assembly Challenges

- Lots of short reads that are not ordered in any way
- Relatively higher error rates versus sequence polymorphisms
- Repetitive regions
- Non-uniform coverage
- Compositional biases

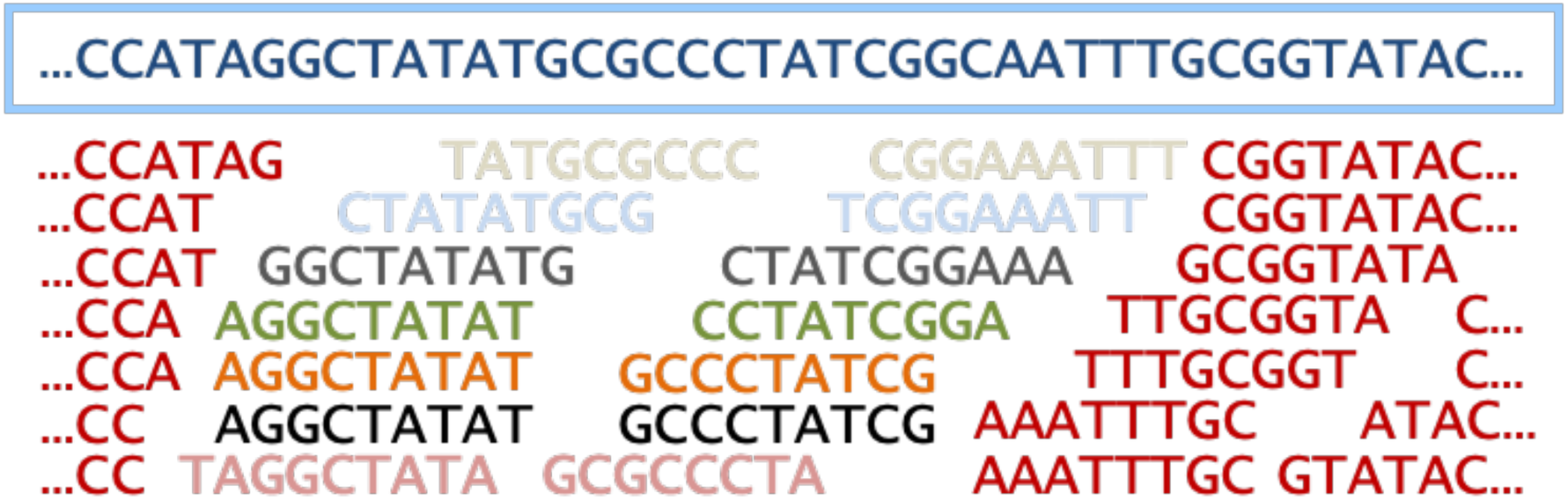
Two Classes of NGS Assembly

1. Alignment-based mapping and assembly: refers to reconstruction of the underlying sequence facilitated by alignments to a previously resolved reference sequences.

2. *De novo* assembly: refers to reconstruction of the underlying sequence without a previously resolved reference sequence

3. Hybrid approach: a combination of the two above

Alignment Mapping and Assembly of Short Reads



- Reads aligned/trimmed to reference genome

Alignment Mapping and Assembly of Short Reads

When a suitable reference sequence is available, index the reference genome sequence and search it efficiently

Generally use a computing strategy called Burrows–Wheeler indexing to notably reduce compute time and memory usage

- **BWA – Burrows-Wheeler Aligner**: a software package for mapping low-divergent sequences against a large reference genome

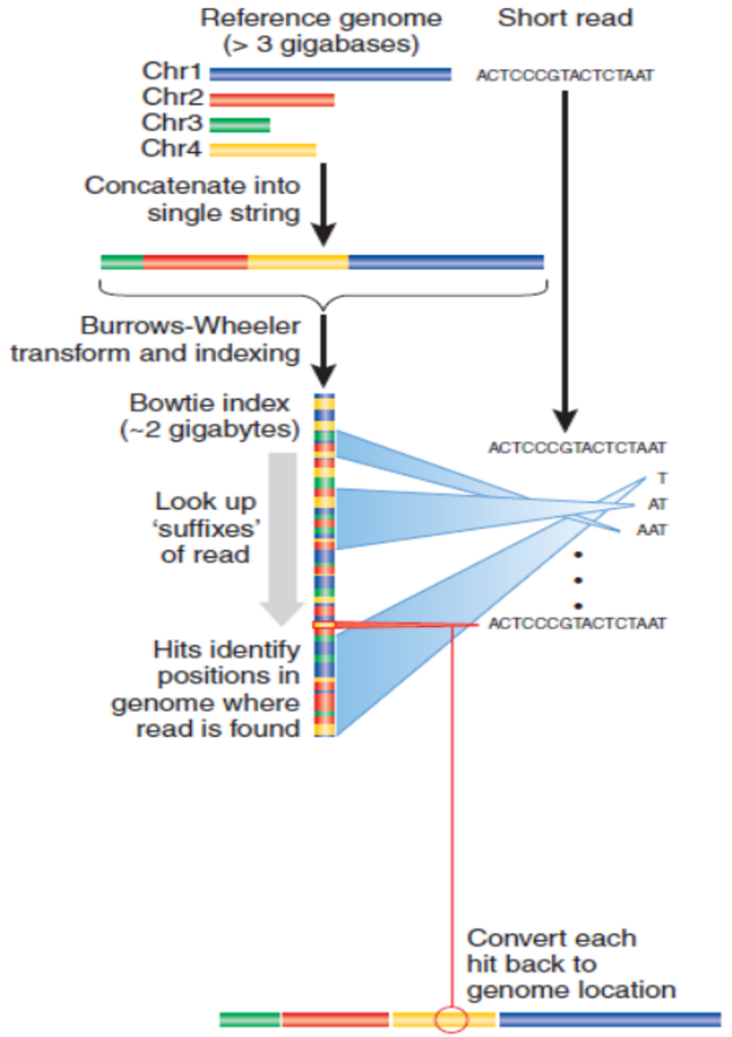
  http://bio-bwa.sourceforge.net

- **Bowtie2** - is an ultrafast and memory-efficient tool for aligning sequencing reads to long reference sequences

  http://bowtie-bio.sourceforge.net/bowtie2/index.shtml

- **SOAPaligner/soap2 (Short Oligonucleotide Analysis Package)**

  https://help.rc.ufl.edu/doc/SOAPaligner



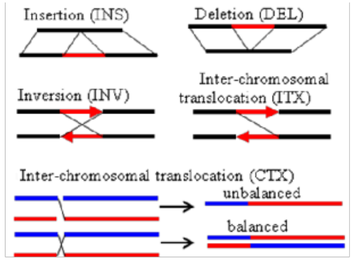
Alignment Mapping and Assembly of Short Reads

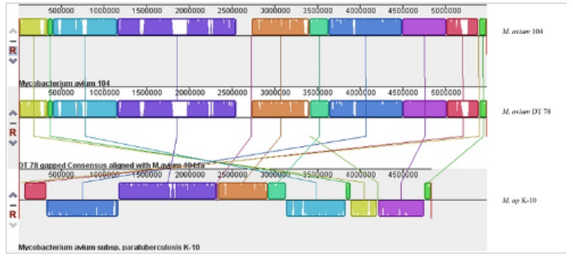
- Mapping to a reference genome requires the same structure of the genome between the template and the re-sequenced organisms

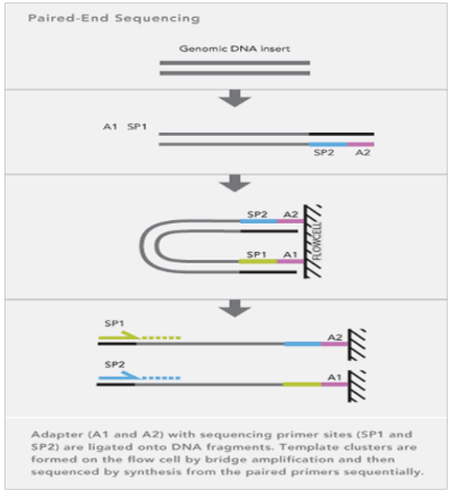
- Small variations like single nucleotide polymorphisms, insertions and deletions are manageable

- Genome rearrangements like repeats, transversions can affect the mapping

  –Paired-end sequencing can help (i.e. reads that are known to be x bases apart)







NIH Fogarty

De novo Assembly of Short Reads

When a suitable reference sequence is not available, assemble reads *de novo*

Generally use a computing strategy called *de Bruijn* graph of k-mers to notably reduce compute time and memory usage; *de Bruijn* graphs are inherently very large due to the observed number of distinct k-mers; require significant computer memory to hold the constructed graph

- **Velvet**

https://www.ebi.ac.uk/~zerbino/velvet/

- **ALLPATHS-LG**

  http://www.broadinstitute.org/software/allpaths-lg/blog/

- **SPAdes**

https://github.com/ablab/spades

Why is *de novo* assembly necessary?

Determine the genome sequence for a “*new*” organism

- No reference available
- Available reference sequence considerably differs from the organism to be studied
- Unbiased view on genome sequence
- Can assemble “multiple genomes” from a set of reads

Hybrid Approach

- Align reads to reference if you can
- *De novo* assemble remaining reads for identification of novel regions/ genomes

Choosing assemblers

- How big is your genome?
- How repetitive?

  ~Short repeats?, Long repeats?, Known repeats?

- Most assemblers are fine tuned for a specific task:

  ~Big mammalian genomes: ALLPATHS, SOAPdenovo, SGA, ABySS

  ~Small genomes: Velvet

  ~Single cell assembly: SPAdes

  ~Transcripts: Trinity, Oases, TrnasABySS, SOAPtrans

Choosing assemblers

- **Public benchmarks**

  # assemblathon.org (compare assemblers for big genomes)

  # gage.cbcb.umd.edu (compares bacterial assemblers)

  # Nucleotid.es provides benchmark data for effectiveness of different assemblers on the same data sets

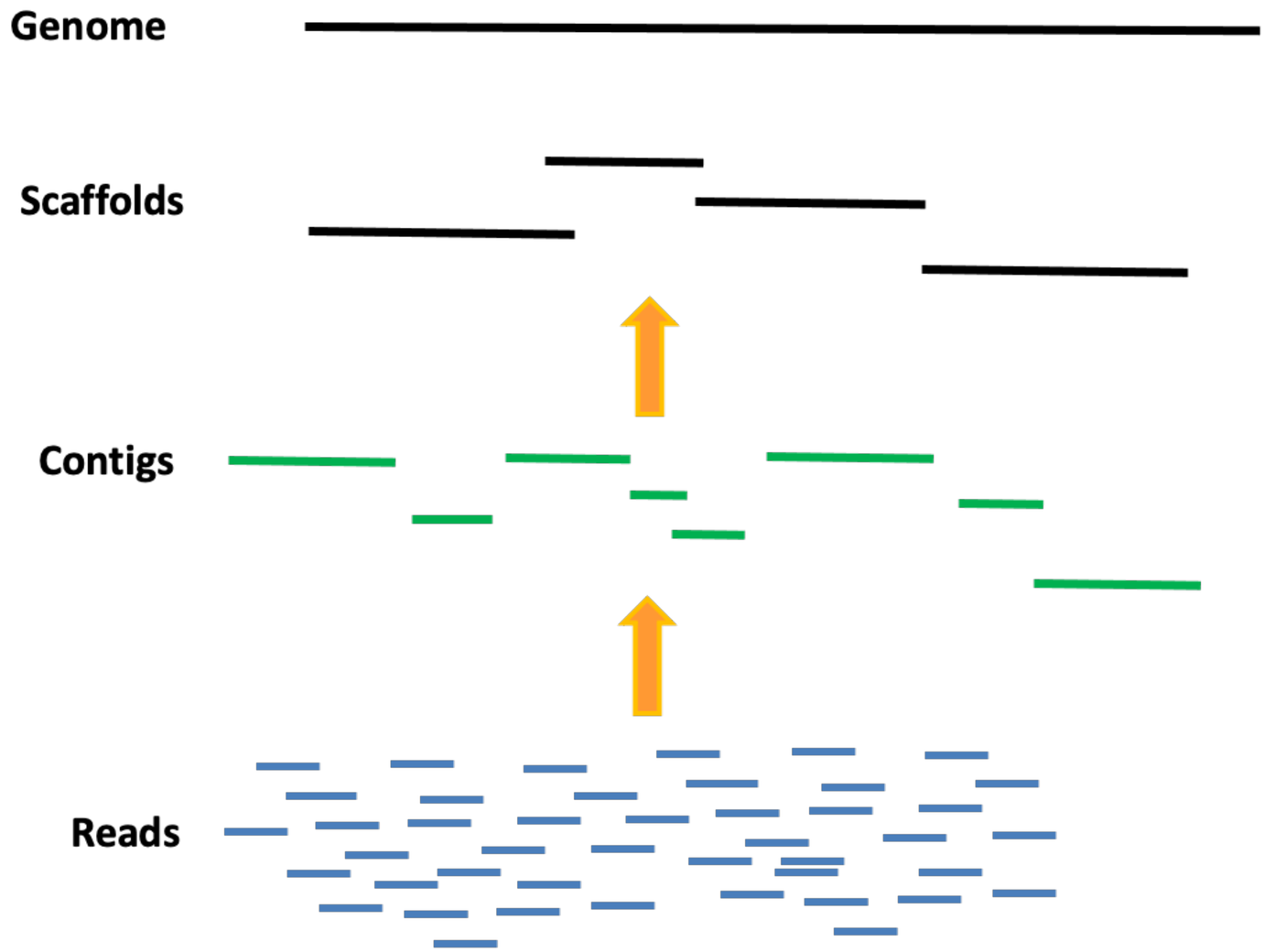
- Setting target coverage will help make things clearer (50 -100X is a good target)

- Choosing the computer hardware

  # Memory (single machine or distributed?)

  # CPU power? Rent cloud CPUs?

Assembly Overview



Comparison of Assemblies

Comparison of assemblies of a single-cell sample of *E.coli* (for contigs ≥ 200 bp)

Assembler	No. of contigs	NGA50 (bp)	Largest (bp)	Total (bp)	Genome fraction (%)	No. of misassemblies	No. of complete genes
EULER-SR	610	26 580	140 518	4 306 898	86.54	19	3442
E+V-SC	396	32 051	132 865	4 555 721	93.58	**2**	3816
IDBA-UD	**283**	90 607	**224 018**	4 734 432	95.90	9	4030
SOAPdenovo	817	16 606	87 533	4 183 037	81.36	6	3060
SPAdes	532	**99 913**	211 020	4 975 641	**96.99**	11	**4071**
Velvet	310	22 648	132 865	3 517 182	75.53	**2**	3121
Velvet-SC	617	19 791	121 367	4 556 809	93.31	**2**	3662

The best value for each column is indicated in bold.

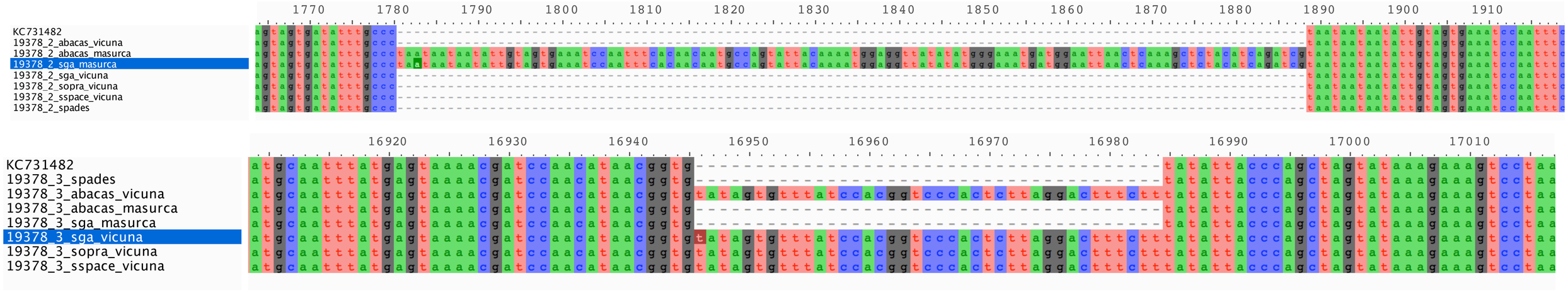
http://quast.sourceforge.net/docs/manual.html#sec3.1

After assembly?

- *ab initio* gene prediction - e.g using glimmer3
- transfer annotations from closely related reference genomes - e.g. using RATT, Geneious
- Call variants
- Phylogenetics
- …all the other fun stuff

Real World Problems

Assembly Problems: Misassemblies

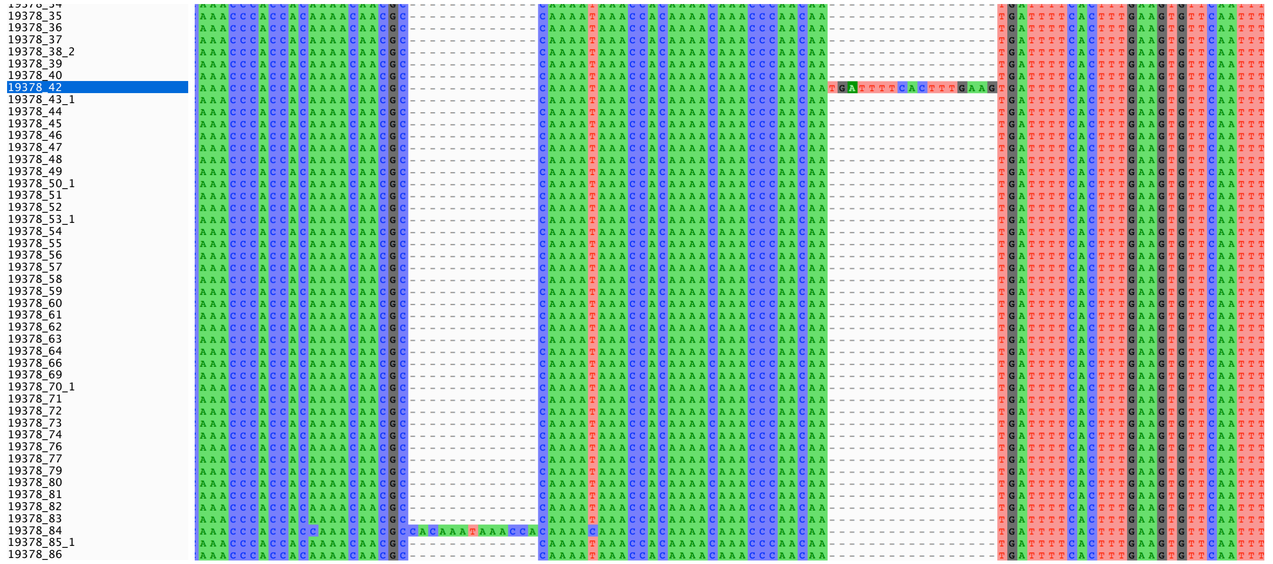


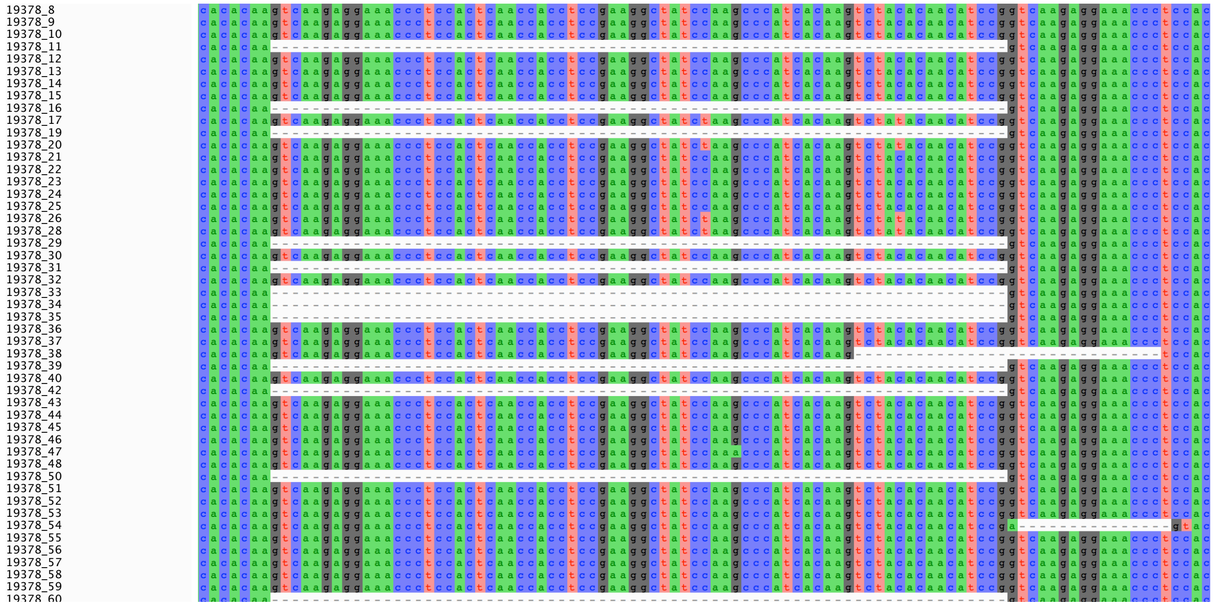
An assembler can perform well in one part of the genome and terribly in another region

Assembly Problems: Misassemblies

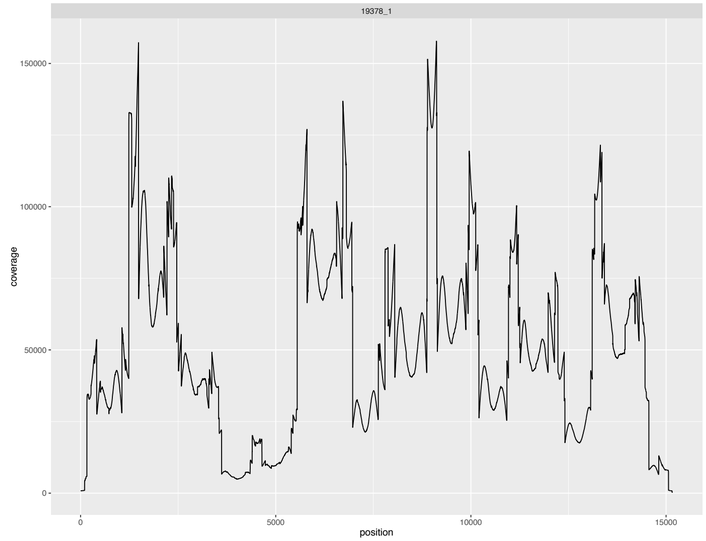
But wait!!!!!!

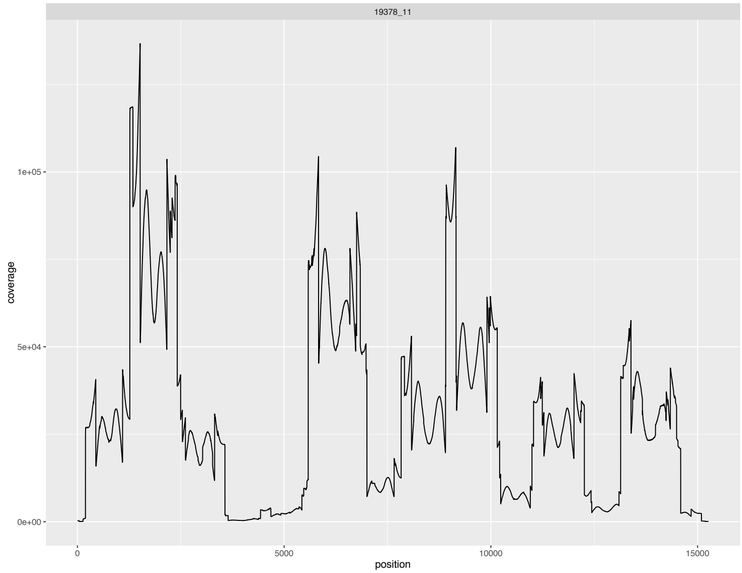
Now it was Spades' turn to perform poorly!

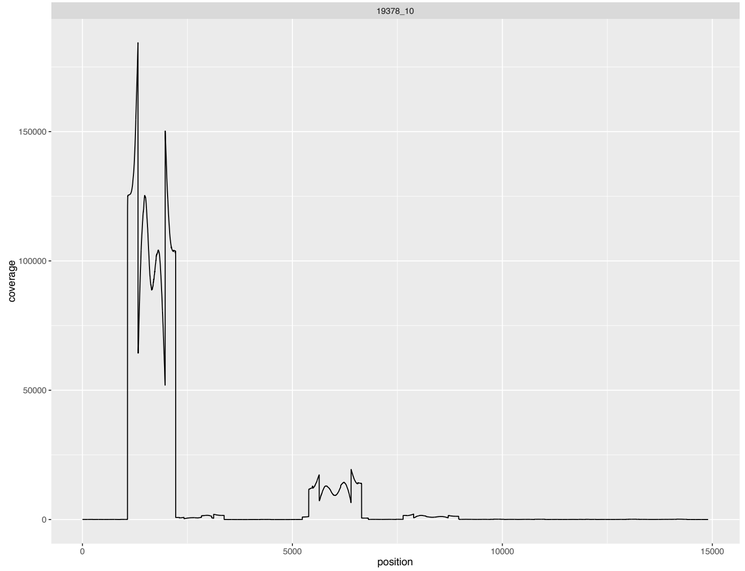


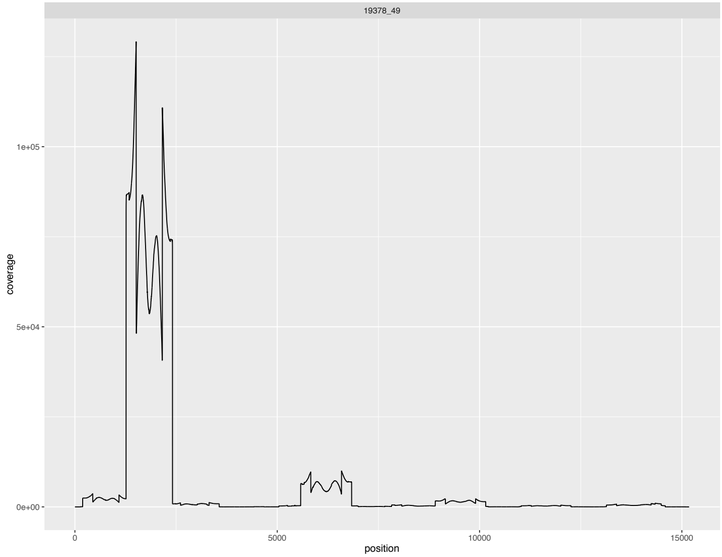


Assembly Problems: Genome Coverage









Assembly problems: Total Reads vs Reads of interest

Sample	raw reads	after QC	after	RSV reads	% RSV	% other	reads	virus	Assembled	Length
19378_1	1,646,861	1,544,831	1,544,531	1,472,516	95.32				viral-ngs	15,168
19378_4	1,109,364	972,030	971,746	503,607	51.81				viral-ngs	14,959
19378_6	1,929,268	1,821,117	1,820,738	1,775,085	97.47				viral-ngs	15,217
19378_9	1,528,436	1,393,891	1,393,574	1,039,271	74.56				viral-ngs	15,140
19378_10	615,019	494,258	493,985	270,736	54.78				viral-ngs	14,900
19378_11	1,095,861	1,002,149	1,001,888	880,695	87.88				spades	15,264
19378_15	1,617,566	1,504,758	1,504,378	1,277,451	84.89				viral-ngs	15,138
19378_16	1,884,316	1,751,871	1,751,431	1,543,592	88.11				N/A	
19378_17	1,467,910	1,271,397	1,264,476	836,047	65.76				viral-ngs	15,049
19378_18	537,251	395,191	394,932	459	0.12				N/A	
19378_19	1,156,299	1,062,937	1,062,709	899,096	84.59	6.2	65943	Betacorona	spades	15,106
19378_21	1,235,013	1,159,498	1,159,243	1,110,734	95.79				viral-ngs	15,188
19378_26	1,262,821	1,183,157	1,182,920	1,132,927	95.75				viral-ngs	15,207
19378_27	1,191,346	940,524	939,991	257,291	27.36				N/A	
19378_28	1,079,769	978,278	978,030	727,601	74.38				viral-ngs	15,049
19378_29	1,337,802	1,246,795	1,246,529	1,189,558	95.41				spades	15,277
19378_30	563,710	451,188	450,949	126,931	28.13				viral-ngs	14,959
19378_31	1,097,173	1,013,483	1,013,222	881,402	86.97				spades	15,276
19378_36	893,343	807,989	807,803	690,282	85.43				viral-ngs	15,074
19378_37	1,048,435	962,762	962,540	903,511	93.85				viral-ngs	15,151
19378_38	849,823	754,392	754,181	584,830	77.52	0.01	50		spades	15,328
19378_39	1,030,430	950,173	949,931	890,375	93.71				spades	15,350
19378_40	816,196	644,925	644,614	180,464	27.98				viral-ngs	14,959
19378_41	772,142	590,587	590,044	111	0.02	1.12	6624	Human	N/A	
19378_42	1,137,914	1,056,339	1,056,097	977,396	92.53				spades	15,282
19378_43	1,143,953	1,052,014	1,051,772	958,567	91.12				viral-ngs	15,049
19378_47	1,179,829	1,113,222	1,113,003	1,086,062	97.56				viral-ngs	15,198
19378_48	1,030,489	949,445	949,171	747,049	78.68				viral-ngs	15,025
19378_49	591,060	478,953	478,748	194,290	40.57				spades	15,179
19378_50	1,173,092	1,092,125	1,091,898	1,064,596	97.48				spades	15,260
19378_51	1,084,795	1,004,037	1,003,797	943,259	93.95				viral-ngs	15,104
19378_52	1,348,996	1,264,645	1,264,364	1,239,457	98.01				viral-ngs	15,183
19378_53	1,349,184	1,235,164	1,234,896	1,105,587	89.51				spades	15,392
19378_54	1,335,970	1,255,716	1,255,472	1,191,208	94.86				spades	15,273
19378_55	1,327,472	1,238,621	1,238,372	1,199,540	96.84				viral-ngs	15,183
19378_59	1,183,503	1,110,963	1,110,733	1,053,090	94.79				viral-ngs	15,139
19378_60	1,276,697	1,181,349	1,181,119	1,119,469	94.76				spades	15,201
19378_86	1 199,614	1,088,365	1088111	960,710	92.12					15,136

NIH Fogarty

Revisiting Shaun Aron's Presentation

Remember:

Conclusions

- Sequence assembly is not a trivial task, and not just YOUR problem
- There is no single “perfect” assembler
- Test and pick the best assembler
- Know your gene/genome…very well
- There might be a best assembler for a specific *pathogen* or *group of pathogens*
- There might also be a best assembler for a particular *dataset*
- Whatever you do downstream depends on that assembly: GIGO