File formats and alignment

MSc in Genomic Medicine

Lucy Crooks

25/1/2016

Analysis steps

Alignment

Variant calling

Quality filtering

Identify key variant

Reads are stored in FASTQ format

- FASTQ is like FASTA but includes quality scores
- Paired end reads are in two files: read 1 (_R1) and read 2

- Files are often compressed: you can tell because they end with .gz

FASTQ format

@M00969:31:000000000-A5GV2:1:1106:21539:11519 2:N:0:2
ATTAAATTCTCAAATTTAATTTTGAGAAGGTTGGTAGAATACTCC
+
CCCCCFFFFFFFGGGGGGGGGGGHHGGCHGHHHHGFHHHHHHHHHH

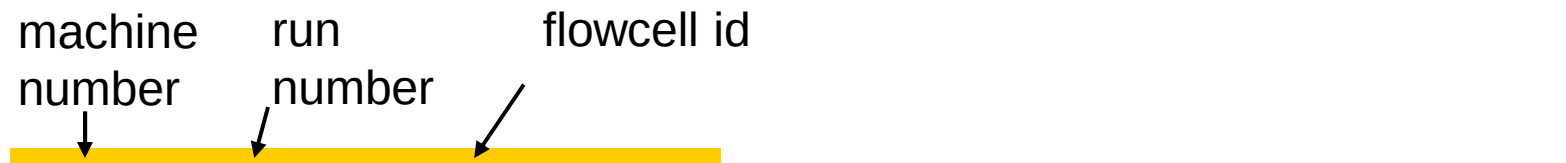
Four lines per read

FASTQ format

First line gives the read id

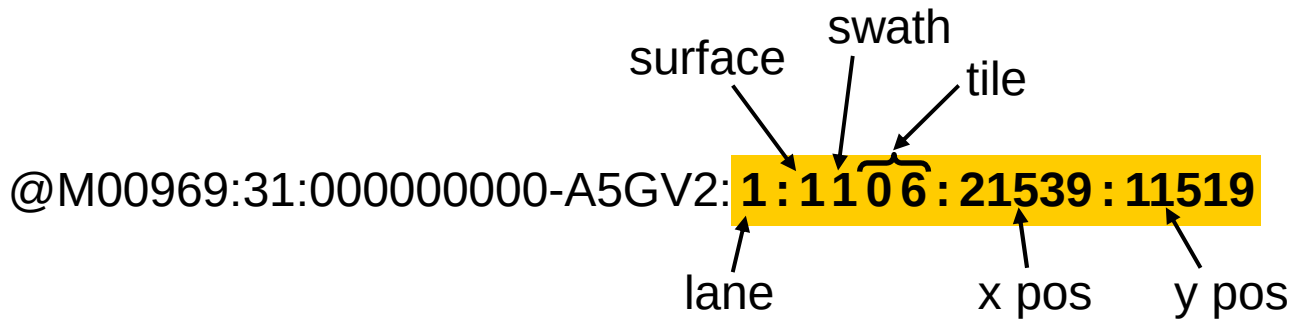
@M00969:31:000000000-A5GV2:1:1106:21539:11519 2:N:0:2
ATTAAATTCTCAAATTTAATTTTGAGAAGGTTGGTAGAATACTCC
+
CCCCCFFFFFFFGGGGGGGGGGHHGGCHGHHHHGFHHHHHHHHHH

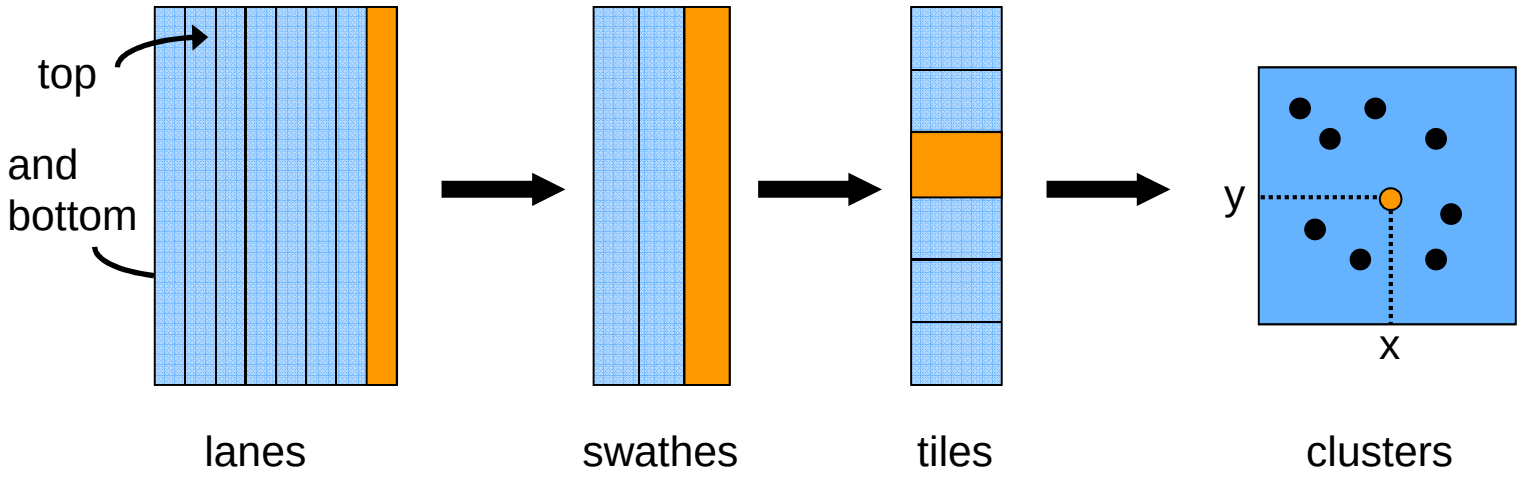
FASTQ format



@M00969:31:000000000-A5GV2:1:1106:21539:11519 2:N:0:2
ATTAAATTCTCAAATTTAATTTTGAGAAGGTTGGTAGAATACTCC
+
CCCCCFFFFFFFGGGGGGGGGGHHGGCHGHHHHGFHHHHHHHHHH

Second part of id gives position on the flow cell





The read id allows you to match read 1 and read 2 that come from the same DNA fragment

FASTQ format

Second line gives the DNA sequence

@M00969:31:000000000-A5GV2:1:1106:21539:11519 2:N:0:2
ATTAAATTCTCAAATTTAATTTTGAGAAGGTTGGTAGAATACTCC
+
CCCCCFFFFFFFGGGGGGGGGGGHHGGCHGHHHHGFHHHHHHHHHH

Written 5′ to 3′ as it is comes off the machine

Could be on the forward or reverse strand

FASTQ format

Fourth line gives the quality scores for each base call

@M00969:31:000000000-A5GV2:1:1106:21539:11519 2:N:0:2
ATTAAATTCTCAAATTTAATTTTGAGAAGGTTGGTAGAATACTCC
+
CCCCCFFFFFFFGGGGGGGGGGGHHGGCHGHHHHGFHHHHHHHHHH

Quality scores are written +33 in ASCII

- ASCII are symbols like ! " # $ % & ' (and characters
- You can look the values up in tables
- Subtract 33 and that gives you the quality score

Quality scores are Phred scaled

- Quality scores tell you about the error probability
  - e.g. chance that the base call is wrong
- Higher quality means less error
- Get probability as $10^{-Q/10}$
- Q30 = 10^{-3} = 0.001 = 1 in 1000 chance of error
- In ASCII, Q30 is ?
- Quality scores usually drop at the end of reads

http://www.somewhereville.com/?p=1508

Alignment programs

- BWA
- Novoalign
- Bowtie2
- ELAND
- Mosaik
- SOAP2

BWA

- Widely used
- Allows for mismatches and gaps
- Three different algorithms
- SDGS uses BWA-aln (backtrack)
  - First finds all good matches for read 1 and read 2 separately
  - Matches are given a penalised score
  - Second step pairs reads; position of partner can help resolve mapping

http://bio-bwa.sourceforge.net/

BWA

- Provides a mapping quality score which is Phred scaled
- Reads that map equally well to more than one position are randomly positioned and given a mapping quality of zero

Aligned reads are stored in BAM files

- BAM is a binary format
- Human readable version is SAM –

  Sequence Alignment/Mapped
- BAM can be converted to SAM using SAMtools view command
- Has information on where read mapped, mapping quality and where partner mapped

BAM format

One line per read

Same sequence identifier

M00969:31:000000000-A5GV2:1:1106:21539:11519 163 chr15
86115312 60 151M = 86115317 156
ATTAAATTCTCAAATTTAATTTTGAGAAGGTTGGTAGAATACTCC
CCCCCFFFFFFFGGGGGGGGGGHHGGCHGHHHHGFHHHHHHHHH
HH X0:i:1 X1:i:0 MD:Z:151 PG:Z:MarkDuplicates RG:Z:NDD
XG:i:0 AM:i:37 NM:i:0 SM:i:37 XM:i:0 XO:i:0 MQ:i:60 XT:A:U

BAM format

The most 5′ position of read is given

And of partner

M00969:31:000000000-A5GV2:1:1106:21539:11519 163 chr15 86115312 60 151M = 86115317 156
ATTAAATTCTCAAATTTAATTTTGAGAAGGTTGGTAGAATACTCC
CCCCCFFFFFFFGGGGGGGGGGHHGGCHGHHHHGFHHHHHHHHH
HH X0:i:1 X1:i:0 MD:Z:151 PG:Z:MarkDuplicates RG:Z:NDD
XG:i:0 AM:i:37 NM:i:0 SM:i:37 XM:i:0 XO:i:0 MQ:i:60 XT:A:U

BAM format

Mapping quality is recorded after the position

M00969:31:000000000-A5GV2:1:1106:21539:11519 163 chr15
86115312 60 151M = 86115317 156
ATTAAATTCTCAAATTTAATTTTGAGAAGGTTGGTAGAATACTCC
CCCCCFFFFFFFGGGGGGGGGGHHGGCHGHHHHGFHHHHHHHHH
HH X0:i:1 X1:i:0 MD:Z:151 PG:Z:MarkDuplicates RG:Z:NDD
XG:i:0 AM:i:37 NM:i:0 SM:i:37 XM:i:0 XO:i:0 MQ:i:60 XT:A:U

BAM format

DNA sequence is retained as well as base qualities

BUT all reads now written as they would appear on the forward strand

M00969:31:000000000-A5GV2:1:1106:21539:11519 163 chr15
86115312 60 151M = 86115317 156
ATTAAATTCTCAAATTTAATTTTGAGAAGGTTGGTAGAATACTCC
CCCCCFFFFFFFFGGGGGGGGGGGHHGGCHGHHHHGFHHHHHHHHH
HH X0:i:1 X1:i:0 MD:Z:151 PG:Z:MarkDuplicates RG:Z:NDD
XG:i:0 AM:i:37 NM:i:0 SM:i:37 XM:i:0 XO:i:0 MQ:i:60 XT:A:U

Paired end reads are often on consecutive lines in a sorted BAM

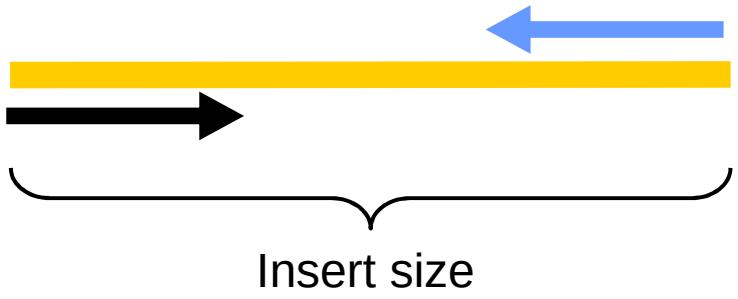
M00969:31:000000000-A5GV2:1:1106:21539:11519 163 chr15 86115312 60 151M = 86115317 156
ATTAAATTCTCAAATTTAATTTTGAGAAGGTTGGTAGAATACTCC

M00969:31:000000000-A5GV2:1:1106:21539:11519 83 chr15 86115317 60 151M = 86115312 -156
ATTCTCAAATTTAATTTTGAGAAGGTTGGTAGAATACTCCATTTC

BAM format

The insert size (fragment length) is reported

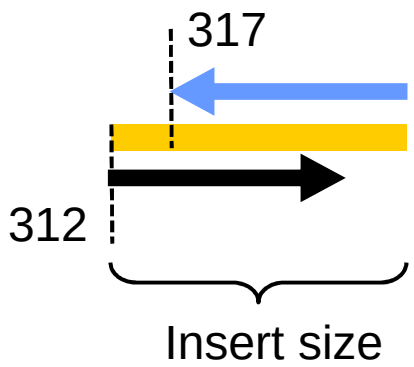
M00969:31:000000000-A5GV2:1:1106:21539:11519 163 chr15
86115312 60 151M = 86115317 156
ATTAAATTCTCAAATTTAATTTTGAGAAGGTTGGTAGAATACTCC
CCCCCFFFFFFFGGGGGGGGGGHHGGCHGHHHHGFHHHHHHHHH
HH X0:i:1 X1:i:0 MD:Z:151 PG:Z:MarkDuplicates RG:Z:NDD
XG:i:0 AM:i:37 NM:i:0 SM:i:37 XM:i:0 XO:i:0 MQ:i:60 XT:A:U



BAM format

The insert size (fragment length) is reported

M00969:31:000000000-A5GV2:1:1106:21539:11519 163 chr15
86115312 60 151M = 86115317 156
ATTAAATTCTCAAATTTAATTTTGAGAAGGTTGGTAGAATACTCC
CCCCCFFFFFFFGGGGGGGGGGHHGGCHGHHHHGFHHHHHHHHH
HH X0:i:1 X1:i:0 MD:Z:151 PG:Z:MarkDuplicates RG:Z:NDD
XG:i:0 AM:i:37 NM:i:0 SM:i:37 XM:i:0 XO:i:0 MQ:i:60 XT:A:U



Read length is 151 bases
5+151=156

Insert size is positive for one read and the same size but negative for the partner

M00969:31:000000000-A5GV2:1:1106:21539:11519 163 chr15 86115312 60 151M = 86115317 156 ATTAAATTCTCAAATTTAATTTTGAGAAGGTTGGTAGAATACTCC

M00969:31:000000000-A5GV2:1:1106:21539:11519 83 chr15 86115317 60 151M = 86115312 -156 ATTCTCAAATTTAATTTTGAGAAGGTTGGTAGAATACTCCATTTC

BAM format

There is a number (flag) summarising the paired alignment

This can be used to filter pairs matching criteria with SAMtools view command

M00969:31:000000000-A5GV2:1:1106:21539:11519 163 chr15
86115312 60 151M = 86115317 156
ATTAAATTCTCAAATTTAATTTTGAGAAGGTTGGTAGAATACTCC
CCCCCFFFFFFFGGGGGGGGGGHHGGCHGHHHHGFHHHHHHHHH
HH X0:i:1 X1:i:0 MD:Z:151 PG:Z:MarkDuplicates RG:Z:NDD
XG:i:0 AM:i:37 NM:i:0 SM:i:37 XM:i:0 XO:i:0 MQ:i:60 XT:A:U

BAM flags

Paired end reads should

- Be from different strands
- Point towards each other

	Paired and both mapped	Proper pair	Read 1	On forward strand	Mate on reverse strand
99	x	x	x	x	x
147	x	x			
83	x	x	x		
163	x	x		x	x

https://broadinstitute.github.io/picard/explain-flags.html

BAM flags

M00969:31:000000000-A5GV2:1:1106:21539:11519 163 chr15 86115312 60 151M = 86115317 156 ATTAAATTCTCAAATTTAATTTTGAGAAGGTTGGTAGAATACTCC

M00969:31:000000000-A5GV2:1:1106:21539:11519 83 chr15 86115317 60 151M = 86115312 -156 ATTCTCAAATTTAATTTTGAGAAGGTTGGTAGAATACTCCATTTC

BAM format

The CIGAR string tells you the length of reads

And if there are gaps in the alignment

M00969:31:000000000-A5GV2:1:1106:21539:11519 163 chr15 86115312 60 151M = 86115317 156 ATTAAATTCTCAAATTTAATTTTGAGAAGGTTGGTAGAATACTCC CCCCCFFFFFFFGGGGGGGGGGGHHGGCHGHHHHGFHHHHHHHHH HH X0:i:1 X1:i:0 MD:Z:151 PG:Z:MarkDuplicates RG:Z:NDD XG:i:0 AM:i:37 NM:i:0 SM:i:37 XM:i:0 XO:i:0 MQ:i:60 XT:A:U

BAM CIGAR string

- Value before M is number of consecutive mapping bases (can be mismatches)
- Value before I is number of bases inserted relative to reference
- Value before D is number of bases deleted relative to reference
- Sum of M and I values equals read length

Example

142M**2I**7M — 2 bp insertion after 142 bases

read length 142+2+7=151

Steps after alignment

- Detecting duplicates
  - Duplicates arise during PCR in library prep
  - Found as pairs which have the same read 1 and read 2 positions
  - Important because fragments are assumed to be independent samples from genome
- Realignment to correct for InDels

Quality Control

- % of target covered at given depth
- % of reads mapping to target
- Genomics England aiming for 95% bases covered at ≥15X with mapping quality ≥ 11X