UCDAVIS Bioinformatics Core

Files and file types

Section 4

Sequencing Read files

fasta files

```
>sequence1
ACCCATGATTTGCGA
```

qual files

```
>sequence1
40 40 39 39 40 39 40 40 40 40 20 20 36 39 39
```

fastq files

```
@sequence1
ACCCATGATTTGCGA
+
IIHHIHIIII55EHH
```

Quality Scores

$$Q = -10log_{10}P$$

Phred Quality Score	Probability of incorrect base call	Base call accuracy
10	1 in 10	90%
20	1 in 100	99%
30	1 in 1000	99.9%
40	1 in 10000	99.99%

Qscore Conversion

$Q_{sanger} = -10log_{10}P$ - based on probability (aka phred)
$Q_{solexa} = -10log_{10}\frac{P}{1-P}$ - based on odds

S - Sanger	Phred+33,	raw reads typically (0, 40)
X - Solexa	Solexa+64,	raw reads typically (-5, 40)
I - Illumina 1.3+	Phred+64,	raw reads typically (0, 40)
J - Illumina 1.5+	Phred+64,	raw reads typically (3, 40)
L - Illumina 1.8+	Phred+33,	raw reads typically (0, 41)

Illumina Read naming conventions

CASAVA 1.8 Read IDs

- @EAS139:136:FC706VJ:2:2104:15343:197393 1:Y:18:ATCACG
  - EAS139 the unique instrument name
  - 136 the run id
  - FC706VJ the flowcell id
  - 2 flowcell lane
  - 2104 tile number within the flowcell lane
  - 15343 'x'-coordinate of the cluster within the tile
  - 197393 'y'-coordinate of the cluster within the tile
  - 1 the member of a pair, 1 or 2 (paired-end or mate-pair reads only)
  - Y Y if the read fails filter (read is bad), N otherwise
  - 18 0 when none of the control bits are on, otherwise it is an even number
  - ATCACG index sequence

SAM/BAM Files

- SAM (Sequence Alignment/Map) format = unified format for storing read alignments to a reference sequence(Consistent since Sept. 2011).
  - http://samtools.github.io/hts-specs/SAMv1.pdf
  - http://samtools.github.io/hts-specs/SAMtags.pdf
- BAM = binary version of SAM for fast querying

SAM/BAM files

SAM files contain two regions

- The header section
  - Each header line begins with character '@' followed by a two-letter record type code
- The alignment section
  - Each alignment line has 11 mandatory fields. These fields always appear in the same order and must be present, but their values can be '0' or '*', if the corresponding information if unavailable, or not applicable.

Sam columns

```
7172283    163    chr9    139389330    60    90M    =    139389482    242    TAGGAGG...    EHHHHHH...
7705896     83    chr9    139389513    60    90M    =    139389512    -91    GCTGGGG...    EBCHHFC...
7705896    163    chr9    139389512    60    90M    =    139389513     91    AGCTGGG...    HHHHHHH...
```

1	QNAME	query template name
2	FLAG	bitwise flag
3	RNAME	reference sequence name
4	POS	1-based leftmost mapping position
5	MAPQ	mapping quality
6	CIGAR	CIGAR string
7	RNEXT	reference name of mate
8	PNEXT	position of mate
9	TLEN	observed template length
10	SEQ	sequence
11	QUAL	ASCII of Phred-scaled base quality

Sam flags

Bit	Description
0x1	template having multiple segments in sequencing
0x2	each segment properly aligned according to the aligner
0x4	segment unmapped
0x8	next segment in the template unmapped
0x10	SEQ being reverse complemented
0x20	SEQ of the next segment in the template being reversed
0x40	the first segment in the template
0x80	the last segment in the template
0x100	secondary alignment
0x200	not passing quality controls
0x400	PCR or optical duplicate
0x800	supplementary alignment

Mapq explained

- MAPQ, contains the "phred-scaled posterior probability that the mapping position" is wrong.
- In a probabilistic view, each read alignment is an estimate of the true alignment and is therefore also a random variable. It can be wrong. The error probability is scaled in the Phred. For example, given 1000 read alignments with mapping quality being 30, one of them will be incorrectly mapped to the wrong location on average.
- A value 255 indicates that the mapping quality is not available.

Mapq explained

- The calculation of mapping qualities is simple, but this simple calculation considers many of the factors below:
  - The repeat structure of the reference. Reads falling in repetitive regions usually get very low mapping quality.
  - The base quality of the read. Low quality means the observed read sequence is possibly wrong, and wrong sequence may lead to a wrong alignment.
  - The sensitivity of the alignment algorithm. The true hit is more likely to be missed by an algorithm with low sensitivity, which also causes mapping errors.
  - Paired end or not. Reads mapped in pairs are more likely to be correct.

Mapq explained

- When you see a read alignment with a mapping quality of 30 or greater, it usually implies:
  - The overall base quality of the read is good.
  - The best alignment has few mismatches.
  - The read has few or just one ‘good’ hit on the reference, which means the current alignment is still the best even if one or two bases are actually mutations, or sequencing errors.

In practice however, each mapper seems to compute the MAPQ in their own way.

Sam cigar

- Compact Idiosyncratic Gapped Alignment Report (CIGAR) SAM flag field:

Op	BAM	Description
M	0	alignment match (can be a sequence match or mismatch)
I	1	insertion to the reference
D	2	deletion from the reference
N	3	skipped region from the reference
S	4	soft clipping (clipped sequences present in SEQ)
H	5	hard clipping (clipped sequences NOT present in SEQ)
P	6	padding (silent deletion from padded reference)
=	7	sequence match
X	8	sequence mismatch

CIGAR Example

	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	2
Ref Pos:	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0
Reference:	C	C	A	T	A	C	T		G	A	A	**C**	**T**	G	A	C	T	A	A	C
Read:					A	C	T	A	G	A	A	**T**	**G**	G		C	T			

POS: 5

CIGAR: 3M1I6M1D2M

** mismatches are not considered in standard CIGAR

GFF/GTF files

- The GFF (General Feature Format) format consists of one line per feature, each containing 9 columns of data (fields). The GTF (General Transfer Format) is identical to GFF version 2.
- Fields must be tab-separated and all fields must contain a value; “empty” fields should be denoted with a ‘.’.
- Columns:
  - Seqname: Name of the sequence chromosome
  - Source: the program, or database, that generated the feature
  - Feature: feature type name, (e.g. gene, exon, cds, etc.)
  - Start: start position of the feature, sequences begin at 1
  - End: stop position of the feature, sequences begin at 1
  - Score: a floating point value (e.g. 0.01)
  - Strand: Defined as ‘+’ (forward),or ‘-’ (reverse)
  - Frame: One of ‘0’, ‘1’, ‘2’, ‘0’ represents the first base of a codon.
  - Attribute: A semicolon-separated list of tag-value pairs, providing additional information about each feature.

GFF/GTF files

Sample GTF output from Ensembl data dump:

```
1 transcribed_unprocessed_pseudogene  gene        11869 14409 . + . gene_id "ENSG00000223972"; gene_name "DDX11L1"; g
1 processed_transcript                transcript  11869 14409 . + . gene_id "ENSG00000223972"; transcript_id "ENST000
```

Sample GFF output from Ensembl export:

```
X       Ensembl Repeat  2419108 2419128 42      .       .       hid=trf; hstart=1; hend=21
X       Ensembl Repeat  2419108 2419410 2502    -       .       hid=AluSx; hstart=1; hend=303
X       Ensembl Repeat  2419108 2419128 0       .       .       hid=dust; hstart=2419108; hend=2419128
X       Ensembl Pred.trans.     2416676 2418760 450.19  -       2       genscan=GENSCAN00000019335
X       Ensembl Variation       2413425 2413425 .       +       .
X       Ensembl Variation       2413805 2413805 .       +       .
```