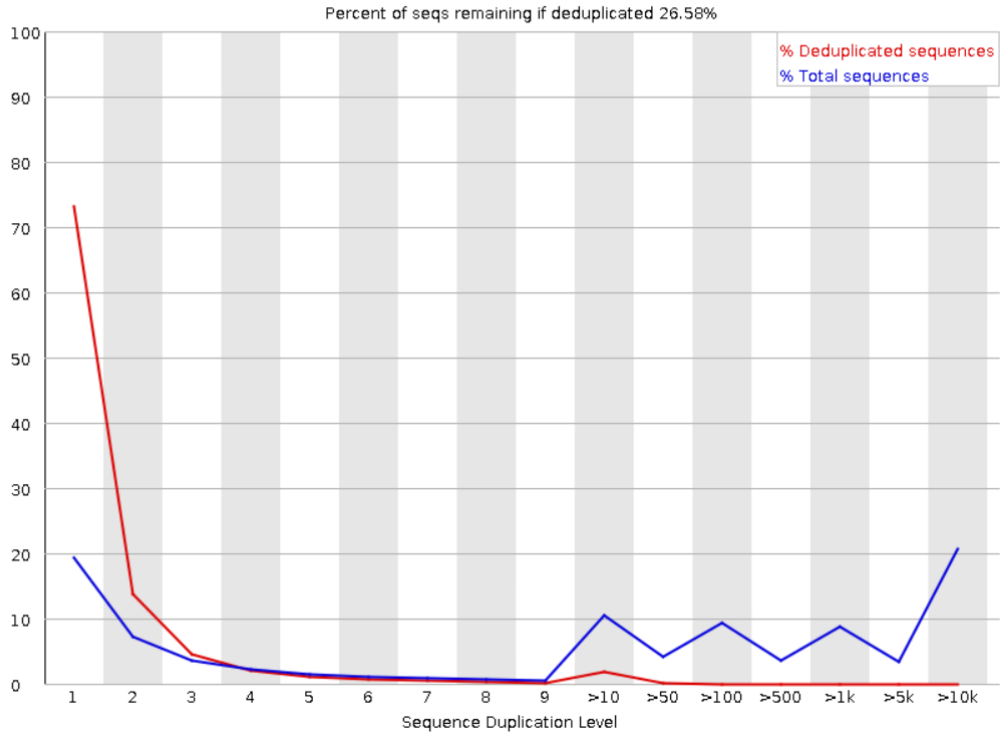
Raw FastQC: Sequence Duplication Levels

Sequence Duplication Levels *Note: These data are based on a sampling of the total number of reads and only the first 50bp are considered

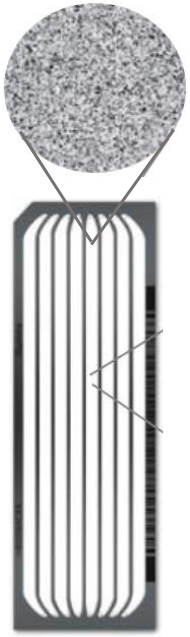


- Used to determine how unique the reads are
- The blue line indicates the percent of duplicated reads at a specific duplication level (red shows what these data would be post deduplication)
- Read duplication can be either biological or technical in origin – How?
- If you have a very diverse library where all reads are unique, 100% of reads would have a duplication level of 1
- It's common, specifically when working with enriched libraries, to have some level of duplication – Why?
- Lower duplication levels (2-9 copies) are likely reads derived from 'interesting' genomic regions (i.e. mRNA)
- Moderate duplication levels (10-100 copies) are likely reads derived from rRNA and/or highly repetitive genomic regions
- High duplication levels (>100 copies) are indicative of a library issue (i.e. adapter dimers; too much PhiX) or contamination

- Are duplicates present in this sample? Should we be concerned? **Yes and yes**
- Roughly what percent of reads are duplicates? **~73.5%**

RNAseq Raw Data – Reasons for Duplications

A Review of Sequencing Duplicate Types



- A single cluster that has falsely been called as two by RTA
- Third party tools may report patterned flow cell clustering duplicates as optical duplicates

Not on Patterned Flow Cells

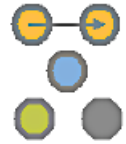
Optical

1 Cluster → Template generation → Called as 2

- Duplicates in nearby wells on Patterned Flow Cells
- During cluster generation a library occupies two adjacent wells

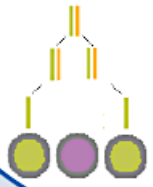
Unique to Patterned Flow Cells

Clustering



- Duplicate molecules that arise from over amplification during sample prep

PCR



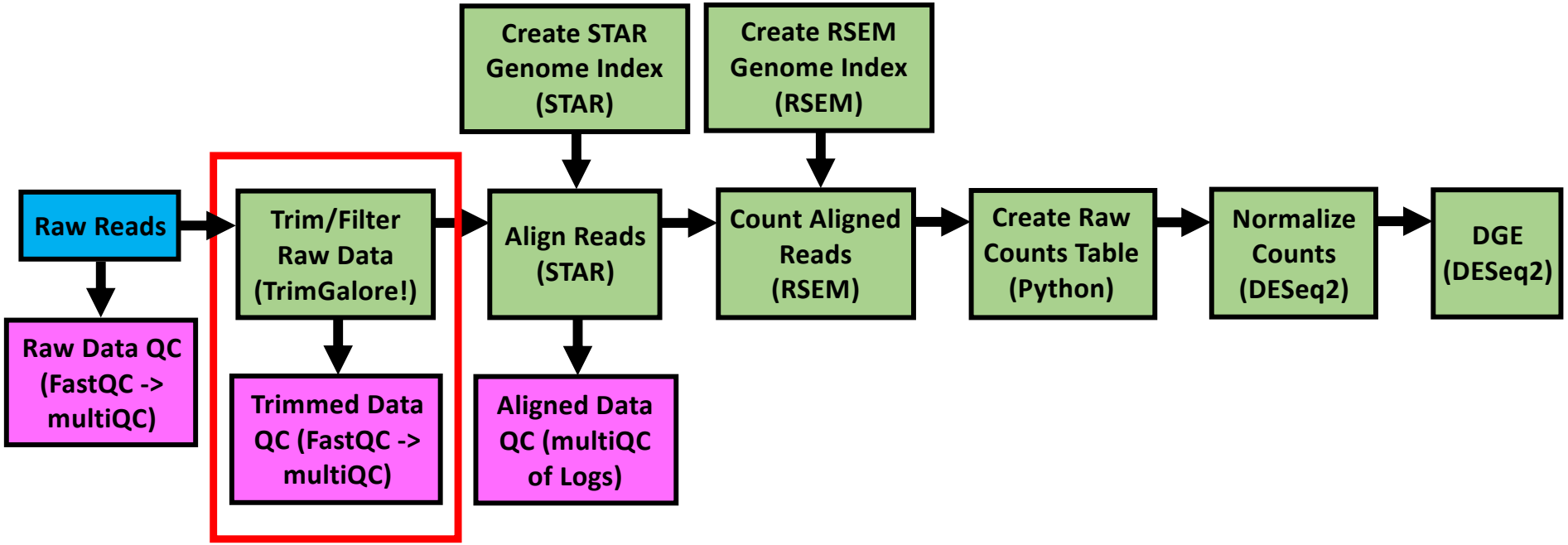
Sister

- Complement strands of same library form independent clusters
- Treated as duplicates by some informatic pipelines

Present on all systems

illumina

RNAseq Workflow: Trimming/Filtering



RNAseq Workflow: Trimming/Filtering

Why would we want to trim/filter (aka pre-process) raw sequence data?

- Remove low quality reads
- Trim adapters
- Trim 3' or 5' end(s)?
- Remove reads that become too short

Preprocessing Tools:

- Cutadapt
  https://cutadapt.readthedocs.io/en/stable/
- **TrimGalore!** (uses cutadapt)
  https://www.bioinformatics.babraham.ac.uk/projects/trim_galore/
- Trimmomatic
  http://www.usadellab.org/cms/?page=trimmomatic
- HTStream
  https://github.com/s4hts/HTStream

TrimGalore! *Parameters:

- --phred33: instructs cutadapt to use ASCII+33 quality scores as Phred scores for quality trimming
- --illumina: defines the adapter sequence to be trimmed as the first 13bp of the Illumina universal adapter AGATCGGAAGAGC
- --quality <INT>: trim low-quality read ends (if not defined, a Phred score cutoff of 20 is applied)
- --length <INT>: remove reads that become shorter than length INT due to quality or adapter trimming (if not defined, a 20bp length threshold is applied)
- --paired: indicates paired-end reads - both reads, forward (R1) and reverse (R2), must pass length threshold or else both reads are removed

*These parameters are specific to the GeneLab standard pipeline, for a complete list of TrimGalore! parameters, go here: https://github.com/FelixKrueger/TrimGalore/blob/master/Docs/Trim_Galore_User_Guide.md#full-list-of-options-for-trim-galore

Trimmed/Filtered Sequence Quality: Basic Stats

Basic Statistics Raw Data

Measure	Value
Filename	Mmus_C57-6J_LVR_RR1_FLT_noERCC_Rep1_M25_R2_raw.fastq.gz
File type	Conventional base calls
Encoding	Sanger / Illumina 1.9
Total Sequences	100600211
Sequences flagged as poor quality	0
Sequence length	150
%GC	52

Basic Statistics Trimmed/Filtered Data

Measure	Value
Filename	Mmus_C57-6J_LVR_RR1_FLT_noERCC_Rep1_M25_R2_trimmed.fastq.gz
File type	Conventional base calls
Encoding	Sanger / Illumina 1.9
Total Sequences	100453545
Sequences flagged as poor quality	0
Sequence length	20-150
%GC	52

- How many reads were removed during pre-processing?

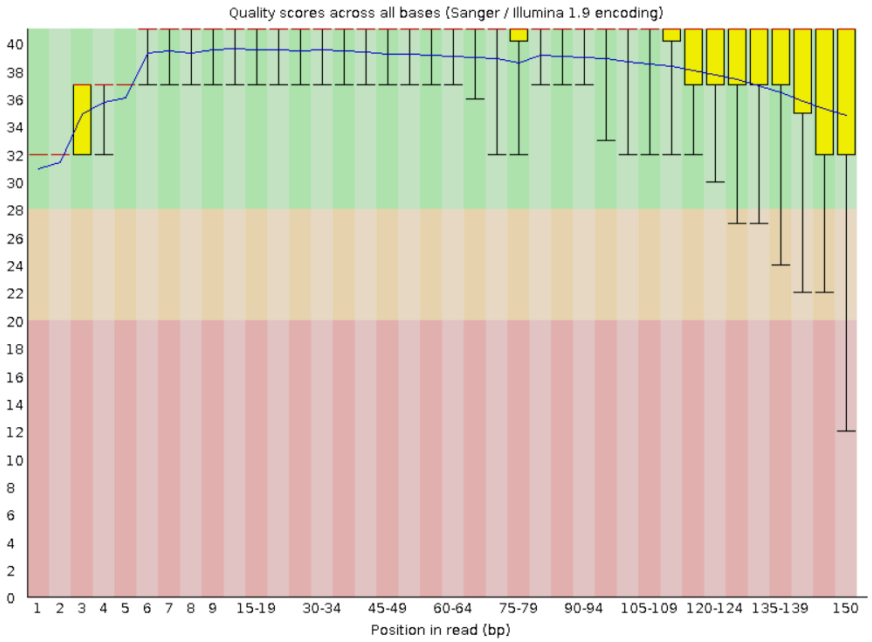
  100,600,211 – 100,453,545 = 146,666

- What is the sample read length after trimming?

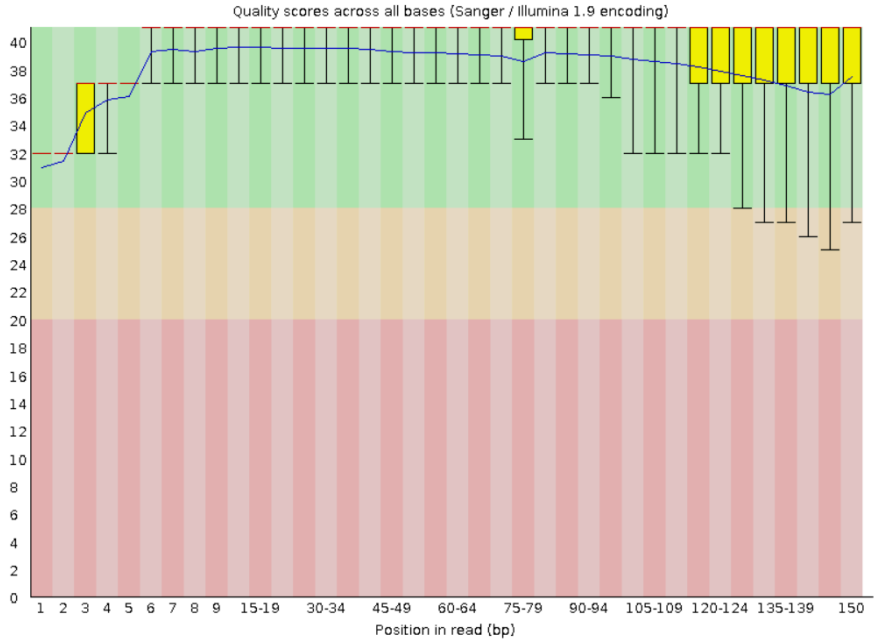
  Ranges from 20 – 150 bp

Trimmed/Filtered Sequence Quality: Per base sequence quality

Per base sequence quality **Raw Data**



Per base sequence quality **Trimmed/Filtered Data**

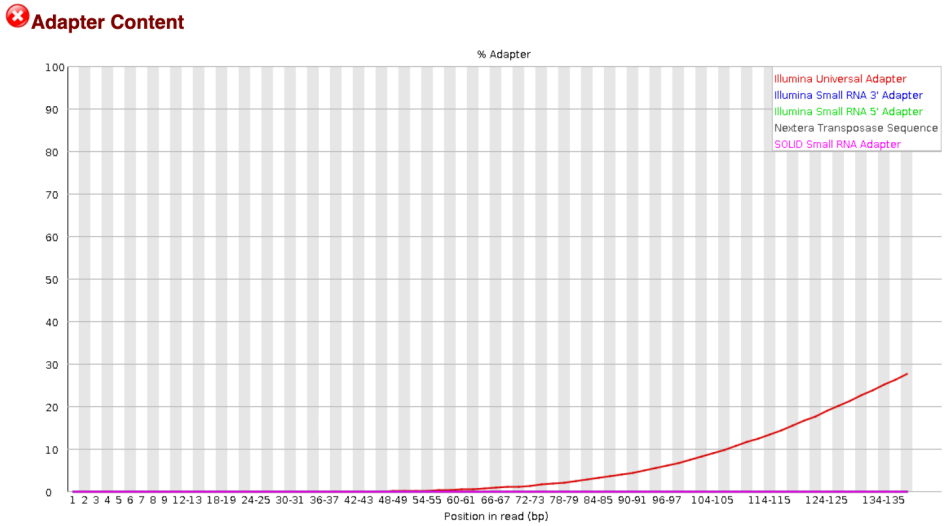


- How has the per-base sequence quality changed after trimming? **It improved!**
- What would happen to the quality if a Phred score cutoff >20 is applied?

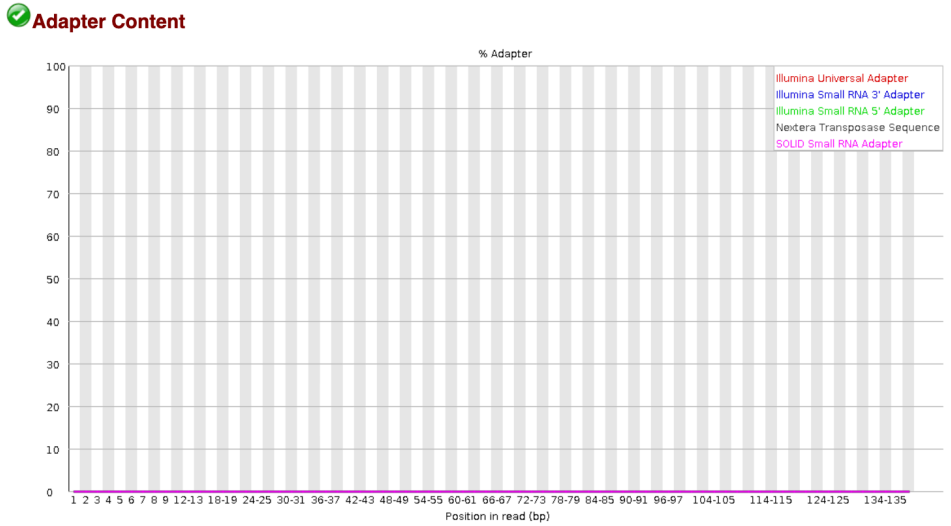
  The post-trimmed reads would have better quality

Trimmed/Filtered Sequence Quality: Adapter Content

Raw Data



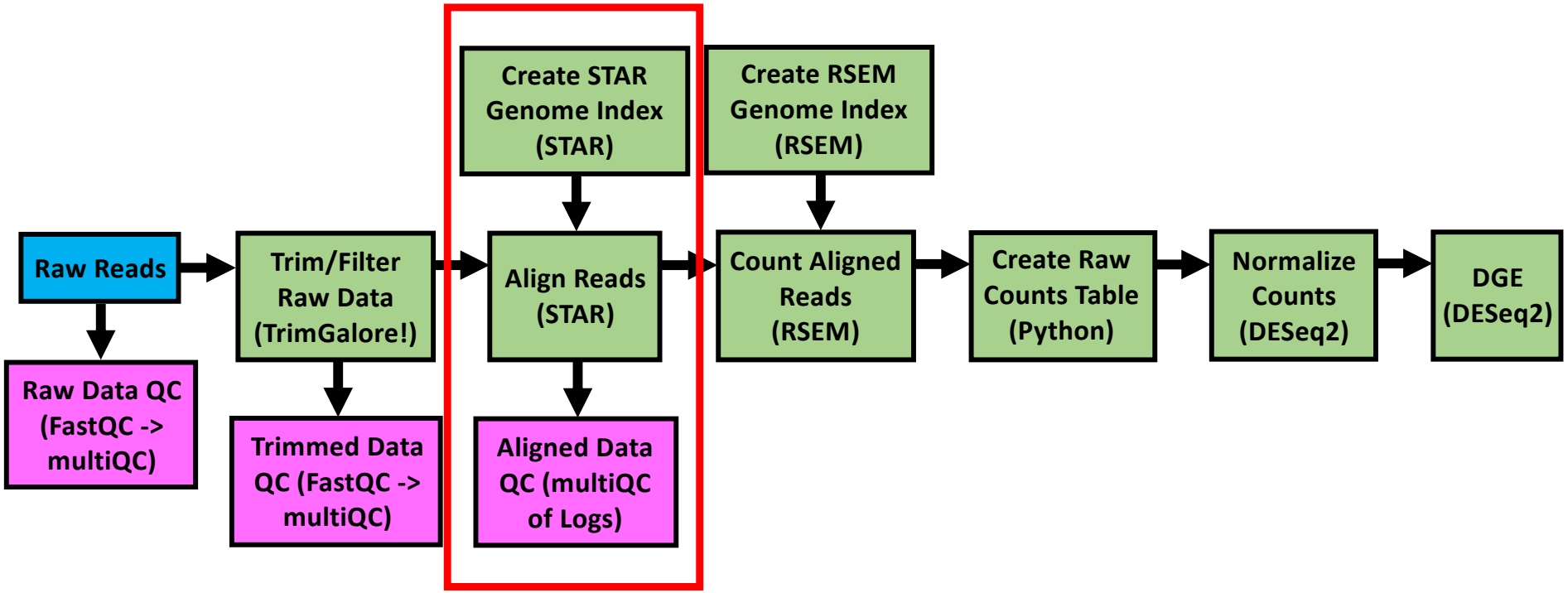
Trimmed/Filtered Data



- What happened to the adapter content after trimming?

Adapters went away!

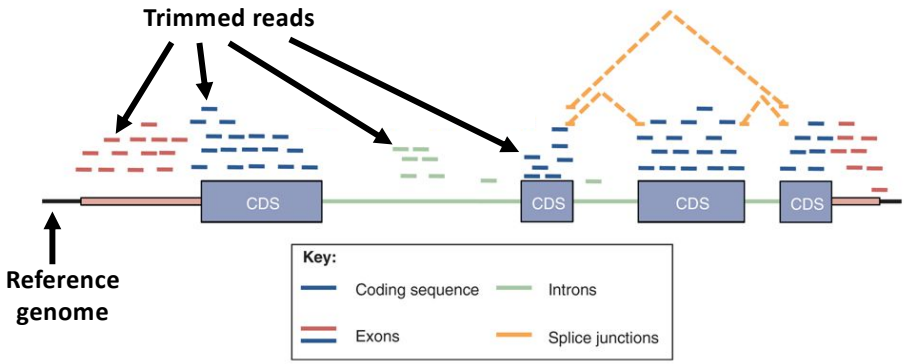
RNAseq Workflow: Alignment



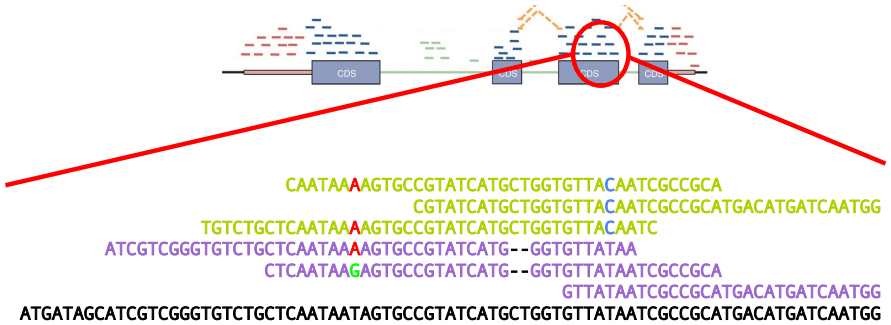
Alignment (aka Mapping?)

- We have millions of short sequences; now how do we figure out where they all came from?
- Luckily, the human genome as well as the genomes of several model organisms have been discovered and these reference genomes are available via public databases – this means we have a place to start looking, a map!

Mapping:



Alignment:



- Mapping reads to a reference genome will tell you where your reads came from (i.e. genomic coordinates)

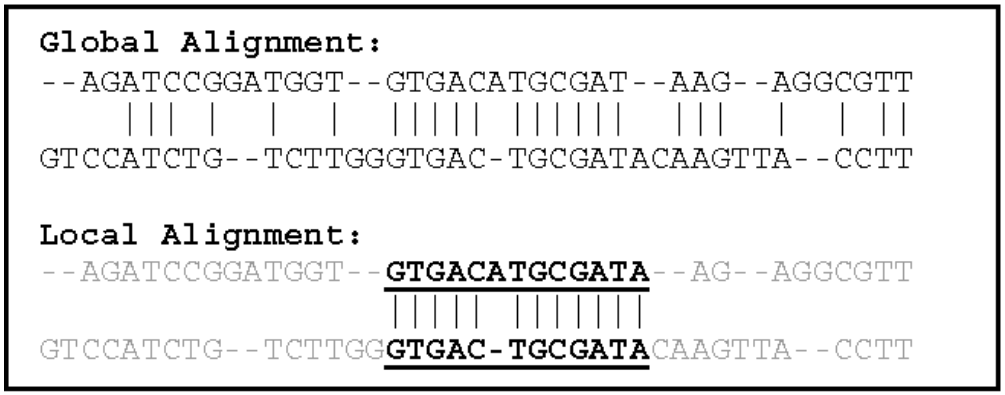
- We may need to fine-tune our mapping to account for differences between our samples and the reference genome
- Aligning reads to a reference genome will also identify single nucleotide differences, gaps, and insertions in addition to the genomic coordinates of origin

Oshlack et al. Genome Biology 2010, 11:220

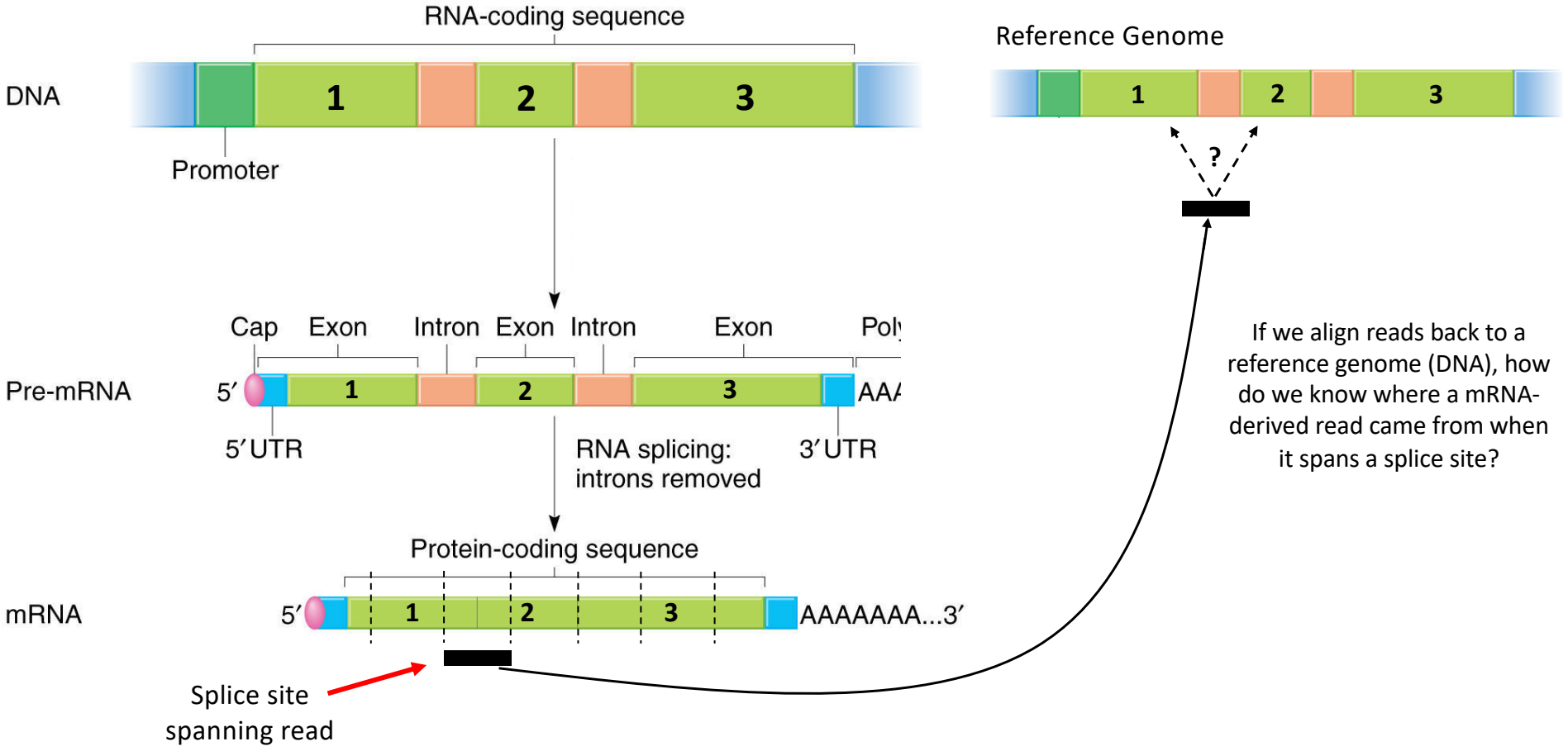
Aligners

- **Global aligners** (Needleman-Wunsch algorithm) – attempt to align the whole provided sequence, end to end, of both the "query" and the "subject/target" (*examples*: aligning two *Pseudomonas* genomes; aligning the mouse and human transcriptomes)
- **Local aligners** (Smith-Waterman algorithm) – attempt to find "hits" or chains of hits within each provided sequence (*example*: identifying genes that share a domain with a target gene)
- **Glocal aligners** – Initial short read aligners assumed that the whole read came from one location within the reference (target) sequence; thus, **glo**bal with respect to the read and lo**cal** with respect to the reference
  - Are there any issues with this approach?

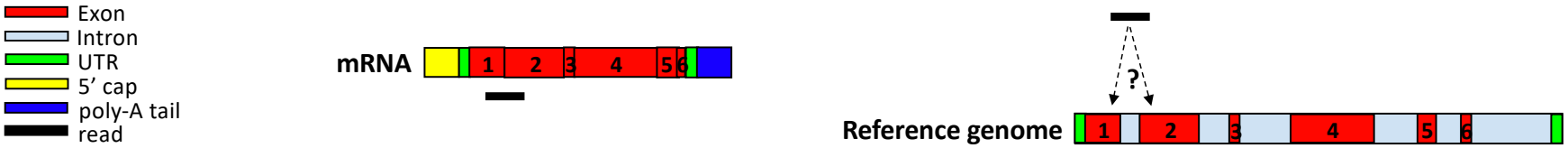
    What if a whole read comes from multiple locations within the reference (spans a splice junction)
- Most aligners commonly used today are local with respect to both the read and reference, which allows them to ignore poor alignment in low quality read ends and/or adapter sequences



Reads That Span Splice Sites



Short Read Aligners – Splice Unaware vs Aware



- **Splice unaware aligners** (Needleman-Wunsch algorithm): Unable to properly align reads that span splice junctions and thus more commonly used for DNA-DNA alignment – Could these be used for aligning RNAseq data?
  - **Pseudo-aligners**: Compares read k-mers (overlapping subsequences) to a transcriptome de Bruijn graph (T-DBG) to find transcripts compatible with the read
- **Splice aware aligners** (Smith-Waterman algorithm): Equipped to handle intron-sized gaps, improving alignment of reads that span splice junctions when aligning to a reference genome and thus are commonly used for transcript-derived cDNA-DNA alignment

Splice unaware aligners:

- Burrows-Wheeler Aligner (BWA)
  - http://bio-bwa.sourceforge.net/
- Bowtie (similar to BWA)
  - http://bowtie-bio.sourceforge.net/bowtie2/index.shtml

Pseudo-aligners:

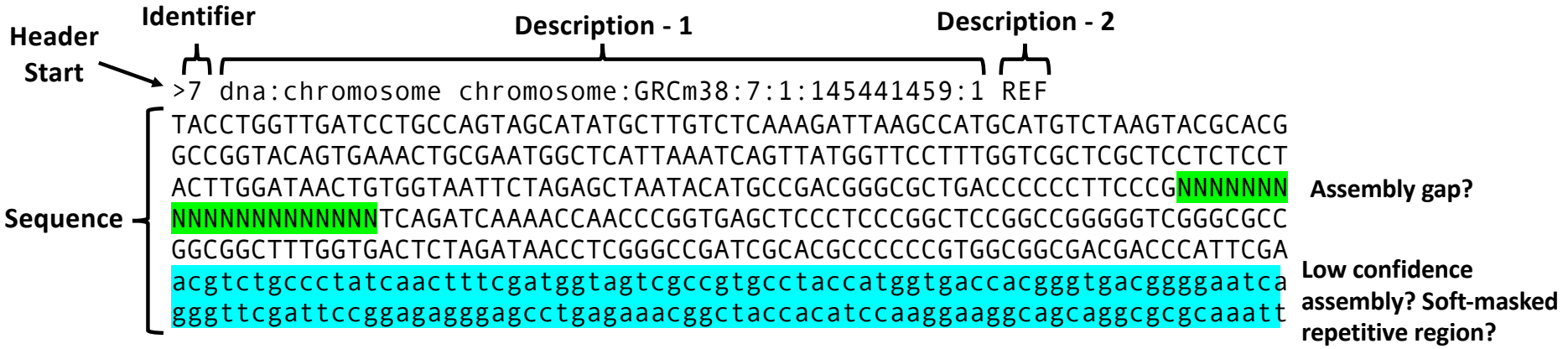
- Kallisto
  - https://pachterlab.github.io/kallisto/
- Salmon
  - https://salmon.readthedocs.io/en/latest/salmon.html

Splice aware aligners:

- **Spliced Transcripts Aligned to a Reference (STAR)**
  - https://github.com/alexdobin/STAR
- Hierarchical indexing for spliced alignment of transcripts 2 (HISAT2)
  - http://daehwankimlab.github.io/hisat2/

Reference Genome (Fasta)

- Short read aligners require a reference – when aligning RNAseq data we use the reference genome of the organism in which the samples were derived
- Reference genomes are stored in fasta files, which have the general format shown below:



- Header Start: All fasta header lines begin with a `>`
- Identifier: Sequence ID – This could be a database-specific ID, a chromosome number (as shown), a gene symbol, etc.
- Description fields: Additional information about the sequence, some databases maintain a standard format for these fields
- Note: Header fields are separated by spaces and some databases also include `|` or `,`

Gene Annotations (GFF/GTF)

- If we want to identify annotated genes (i.e. genes with known genomic coordinates and functions), we need to provide the aligner with a gene annotation file corresponding to reference genome used
  - Gene annotations are stored in Gene Feature Format (GFF) or Gene Transfer Format (GTF) files, which have the following format:

1	2	3	4	5	6	7	8	9
3	ensembl_havana	gene	108107280	108146146	.	-	.	gene_id "ENSMUSG00000000001"; gene_version "4"; gene_name "Gnai3"; gene_source "ensembl_havana";
3	ensembl_havana	transcript	108107280	108146146	.	-	.	gene_id "ENSMUSG00000000001"; gene_version "4"; transcript_id "ENSMUST00000000001";
3	ensembl_havana	exon	108145888	108146146	.	-	.	gene_id "ENSMUSG00000000001"; gene_version "4"; transcript_id "ENSMUST00000000001";
* 3	ensembl_havana	CDS	108145888	108146005	.	-	0	gene_id "ENSMUSG00000000001"; gene_version "4"; transcript_id "ENSMUST00000000001";
3	ensembl_havana	start_codon	108146003	108146005	.	-	0	gene_id "ENSMUSG00000000001"; gene_version "4"; transcript_id "ENSMUST00000000001";

1. **Sequence Name**: Name of the chromosome or scaffold [3]
2. **Source**: Program that generated the GTF file or feature [ensembl_havana]
3. **Feature**: Feature type; gene, exon, CDS, start codon, etc. [CDS]
4. **Start**: Start location on reference sequence [108145888]
5. **End**: End location on reference sequence [108146005]

[*]

6. **Score**: Floating point value [.]
7. **Strand**: Forward (+) or reverse (-) [-]
8. **Frame**: Indicates which base, 0, 1, or 2 is the first base of a codon [0]
9. **Attribute**: `;`-deiminated list of tags with additional info [gene_id "ENSMUSG00000000001"; gene_version "4"; transcript_id "ENSMUST00000000001";]

Where do we get reference genome and annotation files?

- Reference genomes and respective annotation files can be downloaded from publicly available databases:
  - **Ensembl**: https://www.ensembl.org/
  - **Ensembl Genomes**: https://ensemblgenomes.org/
  - GENCODE (uses Ensembl IDs): https://www.gencodegenes.org/
  - Illumina igenomes: https://support.illumina.com/sequencing/sequencing_software/igenome.html
  - NCBI genome: https://www.ncbi.nlm.nih.gov/genome/
  - Specialized databases:
    - https://flybase.org/
    - https://wormbase.org/
    - http://www.xenbase.org/
    - https://vectorbase.org/
    - https://phytozome.jgi.doe.gov/
    - https://www.patricbrc.org/

Building a STAR Index

- Before we can align trimmed reads to a reference genome (or transcriptome), the genome must be indexed
- By creating a genomic (or transcriptomic) index, aligners organize and store the genomic context to make searching the entire genome more efficient

STAR *Parameters:

- --runMode genomeGenerate: Instructs STAR to run genome index generation job to create the STAR indexed reference.
- --genomeSAindexNbases <INT>: Length (in bases) of the SA pre-indexing string, usually between 10 and 15. Longer strings require more memory but allow for faster searches. This value should be scaled down for smaller genomes (like bacteria) to min(14, log2(GenomeLength)/2 - 1). For example, for a 1 megaBase genome this value would be 9.
- --genomeDir: Specifies the path to the directory where the STAR indexed reference will be stored. At least 100GB of available disk space is required for mammalian genomes.
- --genomeFastaFiles: Specifies one or more uncompressed fasta file(s) containg the genome reference sequences.
- --sjdbGTFfile: Specifies the uncompressed file(s) containing annotated transcripts in the standard gtf format.
- --sjdbOverhang <INT>: Indicates the length of the genomic sequence around the annotated junction to be used in constructing the splice junctions database. The length should be one less than the length of the reads.

* These parameters are specific to the GeneLab standard pipeline, for a complete list of STAR parameters, go here: https://github.com/alexdobin/STAR/blob/master/doc/STARmanual.pdf

RNAseq Workflow: Alignment with STAR

STAR *Parameters:

- --twopassMode: Specifies 2-pass mapping mode; the `Basic` option instructs STAR to perform the 1st pass mapping, then automatically extract junctions, insert them into the genome index, and re-map all reads in the 2nd mapping pass.
- --genomeDir: Specifies the path to the directory where the STAR indexed reference is stored.
- --outSAMunmapped: Specifies output of unmapped reads in the SAM format; the `Within` option instructs STAR to output the unmapped reads within the main SAM file.
- --outFilterType: Specifies the type of filtering; the `BySJout` option instructs STAR to keep only those reads that contain junctions that passed filtering in the SJ.out.tab output file.
- --outSAMattributes: List of desired SAM attributes in the order desired for the output SAM file; SAM attribute descriptions can be found here: https://samtools.github.io/hts-specs/SAMtags.pdf
- --outFilterMultimapNmax <INT>: Specifies the maximum number of loci the read is allowed to map to; all alignments will be output only if the read maps to no more loci than this value. <20>
- --outFilterMismatchNmax <INT>: Maximum number of mismatches allowed to be included in the alignment output. <999>
- --outFilterMismatchNoverReadLmax <FLOAT>: Ratio of mismatches to read length allowed to be included in the alignment output; the <0.04> value indicates that up to 4 mismatches are allowed per 100 bases.
- --alignIntronMin <INT>: Minimum intron size; a genomic gap is considered an intron if its length is equal to or greater than this value, otherwise it is considered a deletion. <20>

* These parameters are specific to the GeneLab standard pipeline, for a complete list of STAR parameters, go here: https://github.com/alexdobin/STAR/blob/master/doc/STARmanual.pdf

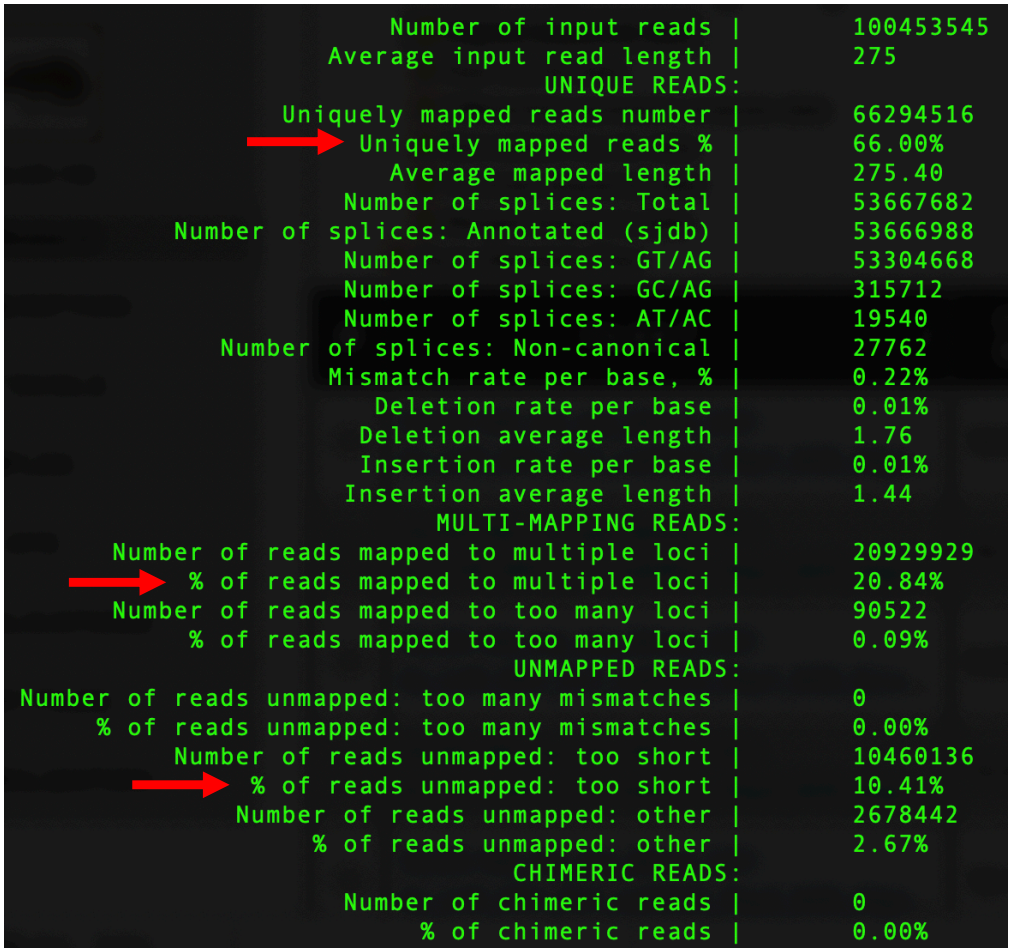
RNAseq Workflow: Alignment with STAR

STAR *Parameters:

- --alignIntronMax <INT>: Maximum intron size. <1000000>
- --alignMatesGapMax <INT>: Maximum genomic distance (in bases) between two mates of paired-end reads; this option should be removed for single-end read. <1000000>
- --alignSJoverhangMin <INT>: Minimum overhang (i.e. block size) for unannotated spliced alignments. <8>
- --alignSJDBoverhangMin <INT>: Minimum overhang (i.e. block size) for annotated spliced alignments. <1>
- --sjdbScore <INT>: Additional alignment score for alignments that cross database junctions. <1>
- --outSAMtype: Specifies desired output format; the `BAM SortedByCoordinate` options specify that the output file will be sorted by coordinate and be in the BAM format
- --quantMode: Specifies the type(s) of quantification desired; the `TranscriptomeSAM` option instructs STAR to output a separate sam/bam file containing alignments to the transcriptome.
- --outSAMheaderHD: Indicates a header line for the SAM/BAM file.
- --outFileNamePrefix: Specifies the path to and prefix for the output file names; for GeneLab the prefix is the sample id.
- --readFilesIn: Path to input read 1 (forward read) and read 2 (reverse read); for paired-end reads, read 1 and read 2 should be separated by a space; for single-end reads only read 1 should be indicated.

* These parameters are specific to the GeneLab standard pipeline, for a complete list of STAR parameters, go here: https://github.com/alexdobin/STAR/blob/master/doc/STARmanual.pdf

STAR Alignment Log



```
                        Number of input reads |       100453545
                    Average input read length |       275
                                  UNIQUE READS:
                   Uniquely mapped reads number |       66294516
                        Uniquely mapped reads % |       66.00%
                          Average mapped length |       275.40
                       Number of splices: Total |       53667682
            Number of splices: Annotated (sjdb) |       53666988
                       Number of splices: GT/AG |       53304668
                       Number of splices: GC/AG |       315712
                       Number of splices: AT/AC |       19540
               Number of splices: Non-canonical |       27762
                      Mismatch rate per base, % |       0.22%
                        Deletion rate per base |       0.01%
                       Deletion average length |       1.76
                       Insertion rate per base |       0.01%
                      Insertion average length |       1.44
                            MULTI-MAPPING READS:
         Number of reads mapped to multiple loci |       20929929
              % of reads mapped to multiple loci |       20.84%
         Number of reads mapped to too many loci |       90522
              % of reads mapped to too many loci |       0.09%
                                  UNMAPPED READS:
Number of reads unmapped: too many mismatches |       0
     % of reads unmapped: too many mismatches |       0.00%
          Number of reads unmapped: too short |       10460136
               % of reads unmapped: too short |       10.41%
              Number of reads unmapped: other |       2678442
                   % of reads unmapped: other |       2.67%
                                  CHIMERIC READS:
                        Number of chimeric reads |       0
                             % of chimeric reads |       0.00%
```

Alignment File (SAM/BAM)

- Alignment data are stored in Sequence Alignment Map (*SAM) files, which have the following format:

1		2	3		4	5	6								
	J00113:339:HMJMNBBXX:3:1101:1144:1367	163	ENSMUST00000031314		1845	255	123M	=	1845	123	NGCAA…GGAAC	#AAFF…JFJJJ	NH:i:	HI:i:1	MC:Z:123M
*	J00113:339:HMJMNBBXX:3:1101:1144:1367	83	ENSMUST00000031314		1845	255	123M	=	1845	-123	TGCAA…GGANC	JFJFJ…FFA#A	NH:i:	HI:i:1	MC:Z:123M
	J00113:339:HMJMNBBXX:3:1101:5426:1367	419	ENSMUST00000174924		135	3	148M	=	144	159	NGTGA…AACAT	#AAFF…JJJJJ	NH:i:2	HI:i:1	MC:Z:150M
	J00113:339:HMJMNBBXX:3:1101:5426:1367	339	ENSMUST00000174924		144	3	150M	=	135	-159	CCGGG…ACCNC	JJJJJ…FFA#A	NH:i:2	HI:i:1	MC:Z:148M
	J00113:339:HMJMNBBXX:3:1101:5426:1367	163	ENSMUST00000175032		135	3	148M	=	144	159	NGTGA…AACAT	#AAFF…JJJJJ	NH:i:2	HI:i:2	MC:Z:150M
	J00113:339:HMJMNBBXX:3:1101:5426:1367	83	ENSMUST00000175032		144	3	150M	=	135	-159	CCGGG…ACCNC	JJJJJ…FFA#A	NH:i:2	HI:i:2	MC:Z:148M

Column	Name	Description [*]
1	QNAME	Query template name: Info about the sequencing run that generated the read, found in line 1 of the trimmed fastq file [J00113:339:HMJMNBBXX:3:1101:1144:1367]
2	FLAG	bitwise FLAG: Information about the alignment encoded in bits. To easily decode the SAM FLAG, type it into the Broad Institute's SAM FLAG decoder: https://broadinstitute.github.io/picard/explain-flags.html [83]
3	RNAME	Reference sequence name: Name of the reference sequence the read aligned to (this will be the ensembl transcript ID in the transcript-aligned BAM file as shown above, and the chromosome number in the genome-aligned BAM file) [ENSMUST00000031314]

Column	Name	Description [*]
4	POS	1-based leftmost mapping position: The position on the reference genome in which the left most base of the read aligns. [1845]
5	MAPQ	Mapping quality: Equality to the -10log(base10) of the probability that the mapping position is wrong; a value of [255] indicates the mapping quality is not available.
6	CIGAR	CIGAR string: Aligned read length and associated operation, which encodes information about the alignment relative to the reference (i.e. match/mismatch, insertion/deletion). [123M]

*Binary Alignment Map (BAM) files are compressed SAM files. Detailed descriptions of the (SAM) format specification can be found here: https://samtools.github.io/hts-specs/SAMv1.pdf

Alignment File (SAM/BAM)

- Alignment data are stored in Sequence Alignment Map (*SAM) files, which have the following format:

1		2	3	4	5	6	7	8	9	10	11	12		
	J00113:339:HMJMNBBXX:3:1101:1144:1367	163	ENSMUST00000031314	1845	255	123M	=	1845	123	NGCAA…GGAAC	#AAFF…JFJJJ	NH:i:	HI:i:1	MC:Z:123M
*	J00113:339:HMJMNBBXX:3:1101:1144:1367	83	ENSMUST00000031314	1845	255	123M	=	1845	-123	TGCAA…GGANC	JFJFJ…FFA#A	NH:i:	HI:i:1	MC:Z:123M
	J00113:339:HMJMNBBXX:3:1101:5426:1367	419	ENSMUST00000174924	135	3	148M	=	144	159	NGTGA…AACAT	#AAFF…JJJJJ	NH:i:2	HI:i:1	MC:Z:150M
	J00113:339:HMJMNBBXX:3:1101:5426:1367	339	ENSMUST00000174924	144	3	150M	=	135	-159	CCGGG…ACCNC	JJJJJ…FFA#A	NH:i:2	HI:i:1	MC:Z:148M
	J00113:339:HMJMNBBXX:3:1101:5426:1367	163	ENSMUST00000175032	135	3	148M	=	144	159	NGTGA…AACAT	#AAFF…JJJJJ	NH:i:2	HI:i:2	MC:Z:150M
	J00113:339:HMJMNBBXX:3:1101:5426:1367	83	ENSMUST00000175032	144	3	150M	=	135	-159	CCGGG…ACCNC	JJJJJ…FFA#A	NH:i:2	HI:i:2	MC:Z:148M

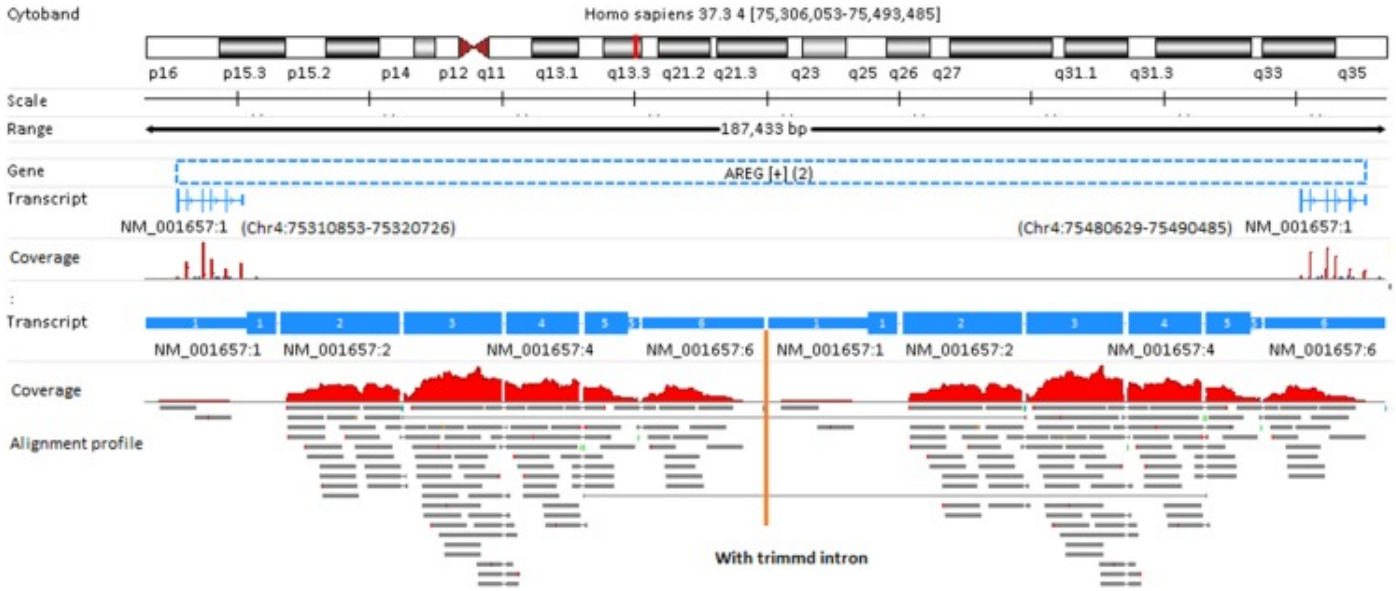
Column	Name	Description [*]
7	RNEXT	Reference name of the mate/next read: Reference sequence name of the next aligned read in the template, if it's the same, this is represented with an equal (=) sign. [=]
8	PNEXT	Position of the mate/next read: 1-based position of the next aligned read in the template. [1845]
9	TLEN	Observed template length: Length from the leftmost position of read 1 to the rightmost position of read 2 for aligned paired-end sequence data. [-123]

Column	Name	Description [*]
10	SEQ	Segment sequence: Sequence of the aligned trimmed read, found in line 2 of the trimmed fastq file. [TGCAA…GGANC]
11	QUAL	ASCII of Phred-scaled base quality +33: Base call quality scores, found in line 4 of the trimmed fastq file. [JFJFJ…FFA#A]
12+	Additional attributes	Additional SAM attributes that were added with the `--outSAMattributes` option in the STAR alignment command. [NH:i: HI:i:1 MC:Z:123M]

*Binary Alignment Map (BAM) files are compressed SAM files. Detailed descriptions of the (SAM) format specification can be found here: https://samtools.github.io/hts-specs/SAMv1.pdf

View Alignment Data

- If you want to take a closer look at how reads map to your reference genome, there are several tools available to help read and interpret BAM/SAM files:
  - Integrate Genomics Viewer (IGV): https://software.broadinstitute.org/software/igv/
  - BAMview: https://www.sanger.ac.uk/tool/bamview/
  - Integrated Genome Browser: https://www.bioviz.org/
  - GenomeView: https://genomeview.org/
  - SAMscope: https://bio.tools/samscope
  - **UCSC Genome Browser**: https://genome.ucsc.edu/



Zhao, Shanrong (2014). PLoS ONE 9(7): e101374. doi:10.1371/ journal.pone.0101374