

# GROUP PRESENTATION

Group 15

Soumik Ray  
Ana Garg  
Swati Gupta

BIOINFORMATICS & ADVANCEMENTS IN AI  
SUPERVISED BY: MS. VARSHINI ARUN

# NGS ANALYSIS

## MYCOBACTERIUM TUBERCULOSIS H37RV

### 1) Sequence ID-

- SRR33053730

### 2) About organism-

- A widely used laboratory strain .
- H37Rv is not a direct clinical isolate but was derived from an earlier strain (H37) that was isolated in 1905.

### 3) Experiment-

- In-vivo screening of M. tuberculosis H37Rv virulence gene in mice.

### 4) Why?

- Leading cause of infectious mortality
- It is a drug-sensitive reference virulent strain
- ~4.4 Mb genome
- Relevance: drug resistance and virulence studies

# STEPS PERFORMED

## MYCOBACTERIUM TUBERCULOSIS H37RV

- 1)** FastQC= To check quality of raw reads by illumina sequencing dataset.
- 2)** Trimmomatic= Trimming of low quality and over-represented reads to enhance the overall quality.
- 3)** BWA alignment= Indexing and aligning the reference sequence with the complete genomic sequence.
- 4)** IGV visualising= Visualizing the mutations or deviations in the bases of our reference sequence from the genomic dataset.

# FASTQC ANALYSIS

## MYCOBACTERIUM TUBERCULOSIS H37RV

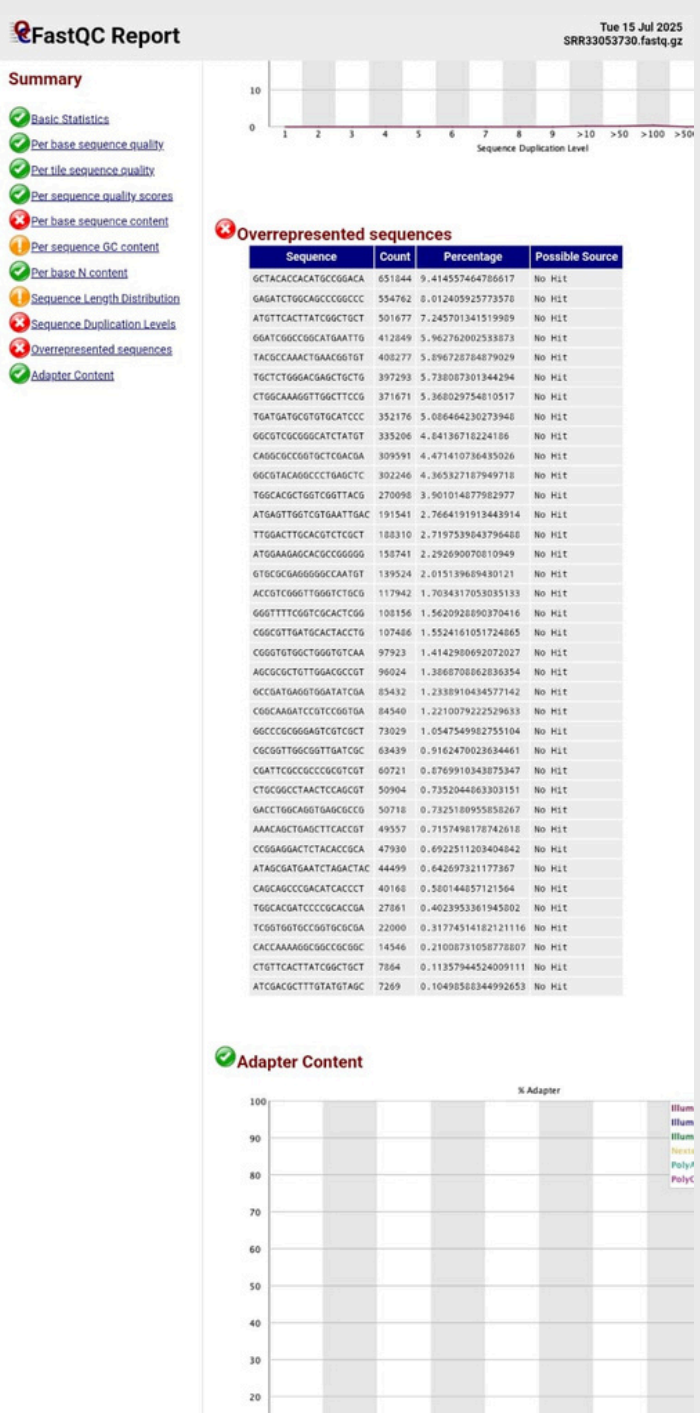
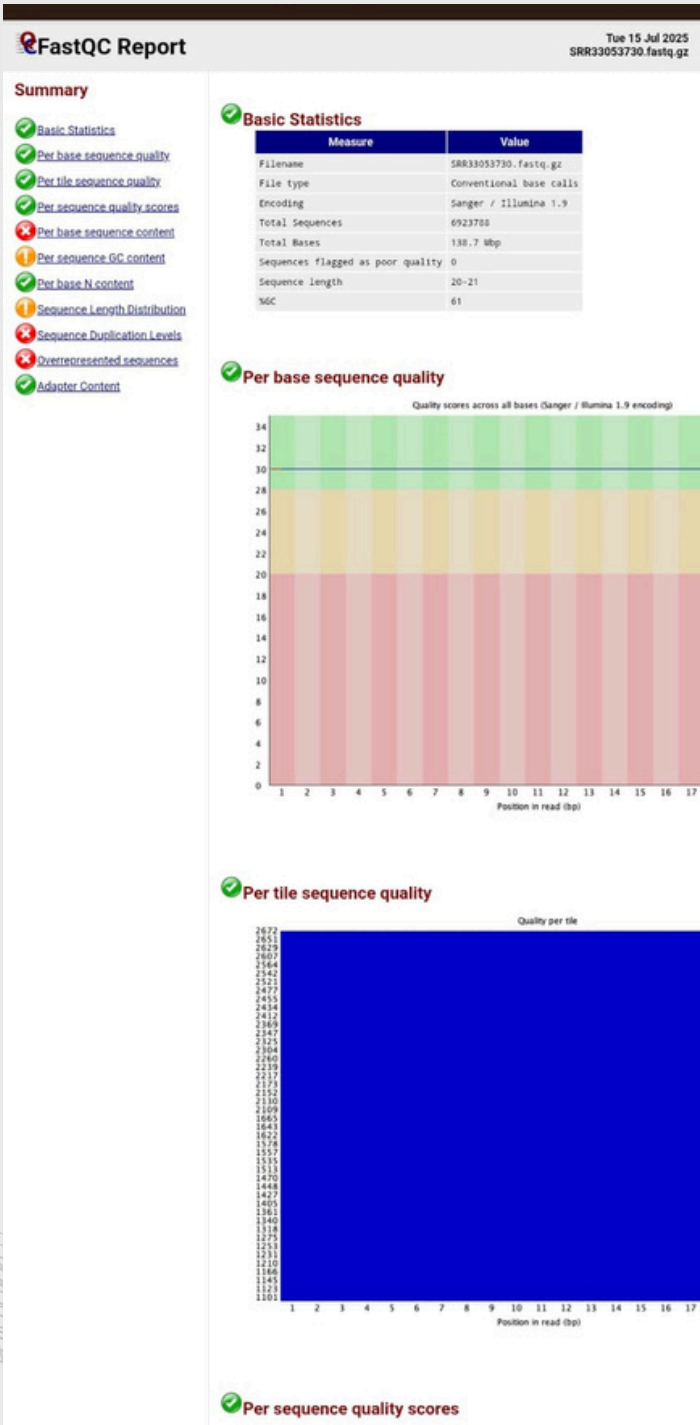
- Total sequences= 6923788
- Total bases= 138.7 Mbp
- % GC content= 61
- Per base sequence quality= excellent as the graph lies in the green region and is uniform.
- Quality score= excellent
- Per sequence quality score= no peaks; excellent
- Per base N content= no addition of unknown bases.





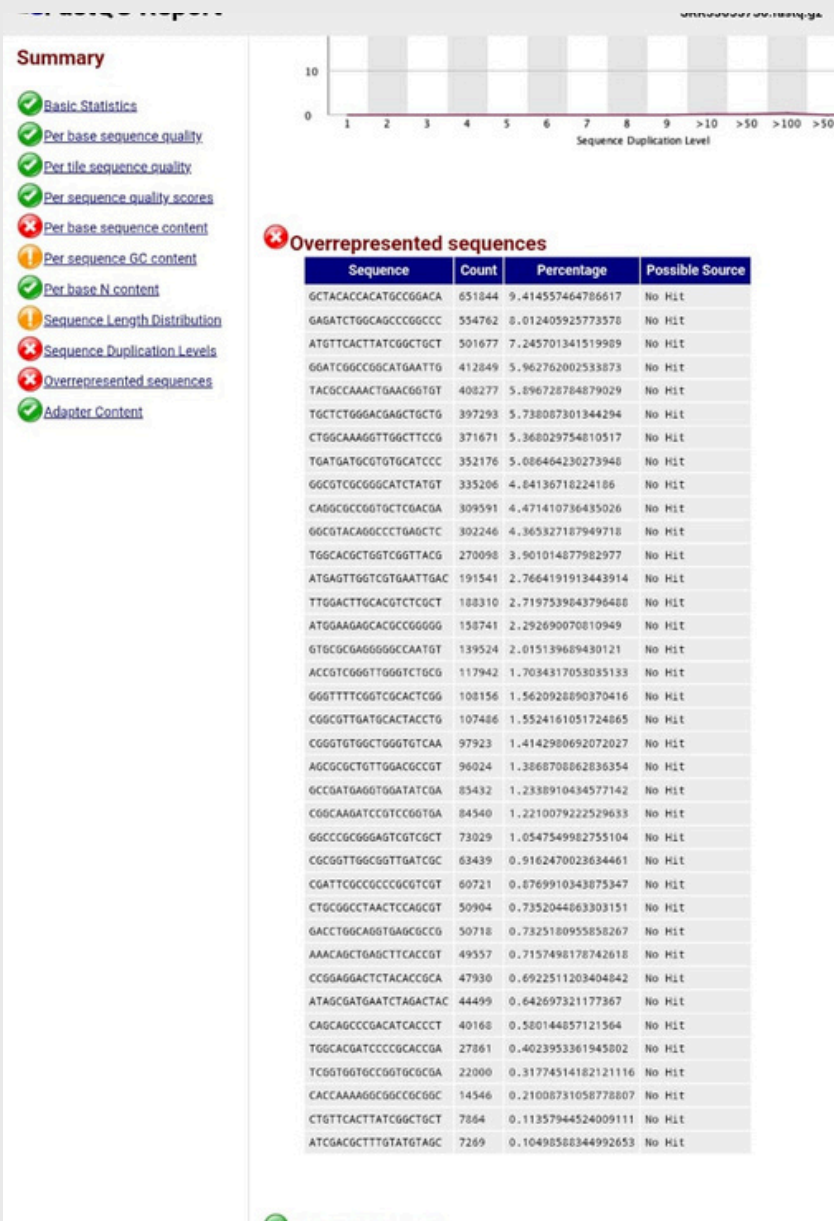
# FASTQC ANALYSIS

## MYCOBACTERIUM TUBERCULOSIS H37RV

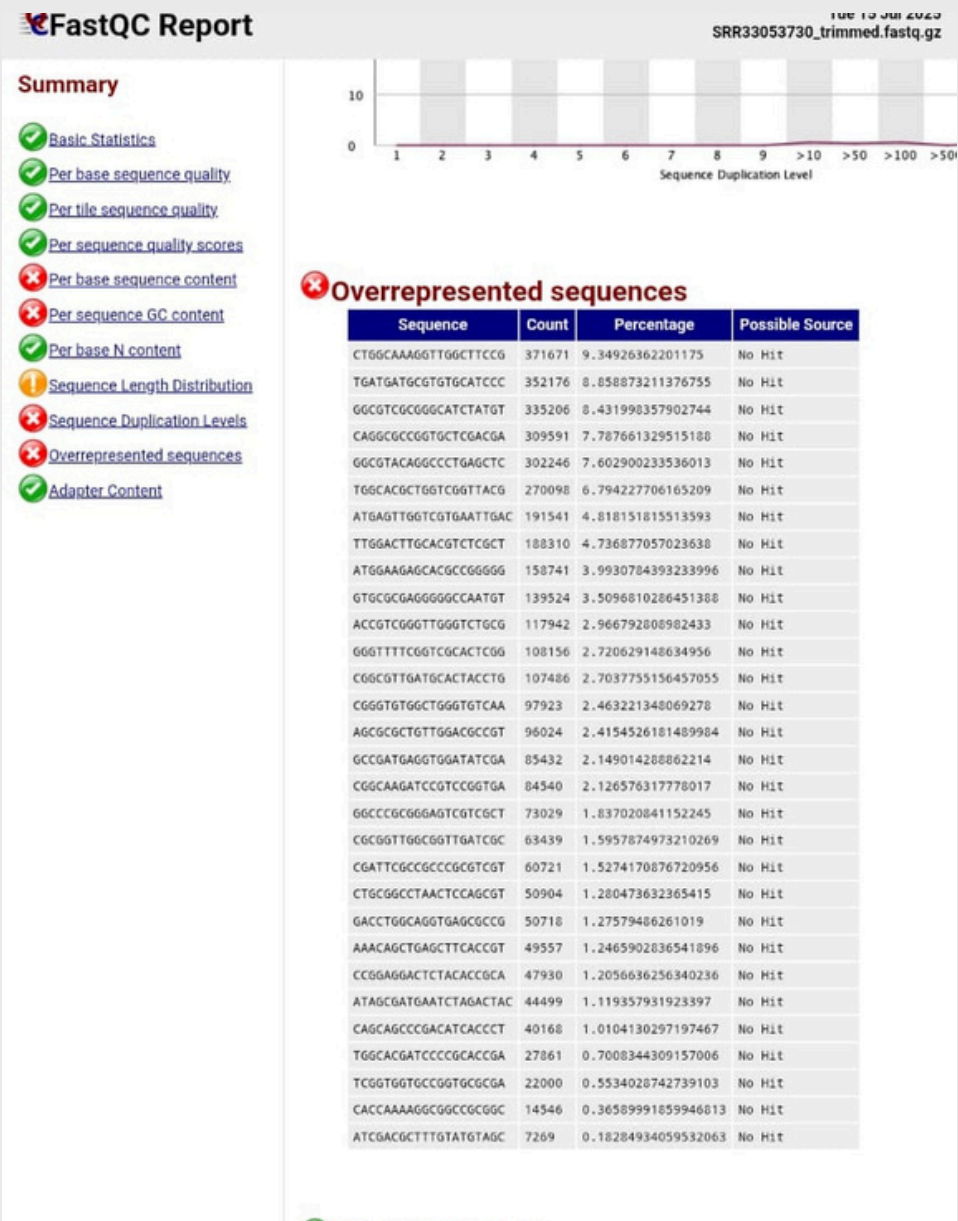


# TRIMMOMATIC ANALYSIS

MYCOBACTERIUM TUBERCULOSIS H37RV



TRIMMING



BEFORE (9.41%)

AFTER (9.34%)

05



# Q&A

## MYCOBACTERIUM TUBERCULOSIS H37RV

### **Why We Picked This Dataset ?**

We chose the dataset SRR33053730 because it felt like a solid fit for our analysis. It was publicly available through SRA, came from a published study, and had good sequencing depth – which is super important for getting reliable results. Plus, it focused on a biological system we were genuinely interested in, so it gave us a good reason to dig deeper and try to understand what's going on.

### **Challenges During FASTQC and Trimming.**

When we first looked at the raw reads using FASTQC, there were a few common problems. Toward the end of the reads, the quality dropped off quite a bit – which happens a lot with Illumina data. We also found adapter sequences still hanging around, and some overrepresented sequences probably coming from primers or technical biases. The trimming step wasn't exactly smooth either. If we trimmed too aggressively, we ended up with really short reads. But if we trimmed too lightly, the low-quality parts stayed in. It took a few rounds of trial and error to get the settings just right so we didn't lose too much data but still cleaned things up properly.

# BWA ALIGNMENT

## MYCOBACTERIUM TUBERCULOSIS H37RV

### Get the reference genome

```
gunzip GCF_000277735.2_ASM27773v2_cds_from_genomic.fna.gz
```

### Index the reference genome

```
bwa index GCF_000277735.2_ASM27773v2_cds_from_genomic.fna
```

### Reads alignment using bwa-mem

```
bwa mem GCF_000277735.2_ASM27773v2_cds_from_genomic.fna  
SRR33053730_trimmed.fastq.gz > aligned.sam
```

### Sam to Bam conversion using sam tools

```
samtools view -Sb aligned.sam > aligned.bam
```

### Sort Bam files

```
samtools sort aligned.bam -o aligned_sorted.bam
```

### Indexing Bam files

```
samtools index aligned_sorted.bam
```



# COMMANDS

## MYCOBACTERIUM TUBERCULOSIS H37RV

```
(base) ~ $ cd Documents/NGS
(base) NGS $ fastqc SRR33053730.fastq.gz

application/gzip
Started analysis of SRR33053730.fastq.gz
Approx 5% complete for SRR33053730.fastq.gz
Approx 10% complete for SRR33053730.fastq.gz
Approx 15% complete for SRR33053730.fastq.gz
Approx 20% complete for SRR33053730.fastq.gz
Approx 25% complete for SRR33053730.fastq.gz
Approx 30% complete for SRR33053730.fastq.gz
Approx 35% complete for SRR33053730.fastq.gz
Approx 40% complete for SRR33053730.fastq.gz
Approx 45% complete for SRR33053730.fastq.gz
Approx 50% complete for SRR33053730.fastq.gz
Approx 55% complete for SRR33053730.fastq.gz
Approx 60% complete for SRR33053730.fastq.gz
Approx 65% complete for SRR33053730.fastq.gz
Approx 70% complete for SRR33053730.fastq.gz
Approx 75% complete for SRR33053730.fastq.gz
Approx 80% complete for SRR33053730.fastq.gz
Approx 85% complete for SRR33053730.fastq.gz
Approx 90% complete for SRR33053730.fastq.gz
Approx 95% complete for SRR33053730.fastq.gz
Analysis complete for SRR33053730.fastq.gz
(base) NGS $ java -jar trimmomatic.jar SE -phred33 SRR33053730.fastq.gz SRR33053730_trimmed.fastq.g
z \
> ILLUMINACLIP:adapter.fa:2:30:10 \
> LEADING:3 TRAILING:3 SLIDINGWINDOW:4:20 MINLEN:15
TrimmomaticSE: Started with arguments:
-phred33 SRR33053730.fastq.gz SRR33053730_trimmed.fastq.gz ILLUMINACLIP:adapter.fa:2:30:10 LEADING
:3 TRAILING:3 SLIDINGWINDOW:4:20 MINLEN:15
Automatically using 4 threads
Using Medium Clipping Sequence: 'GGATCGGCCGGCATGAATTG'
Using Medium Clipping Sequence: 'ATGTTCACTTATCGGCTGCT'
Using Medium Clipping Sequence: 'TGCTCTGGGACGAGCTGCTG'
Using Medium Clipping Sequence: 'TACGCCAAACTGAACGGTGT'
Using Medium Clipping Sequence: 'GAGATCTGGCAGCCCGGCC'
Using Medium Clipping Sequence: 'GCTACACCACATGCCGGACA'
ILLUMINACLIP: Using 0 prefix pairs, 6 forward/reverse sequences, 0 forward only sequences, 0 revers
e only sequences
Input Reads: 6923788 Surviving: 3975404 (57.42%) Dropped: 2948384 (42.58%)
TrimmomaticSE: Completed successfully
(base) NGS $ fastqc SRR33053730_trimmed.fastq.gz

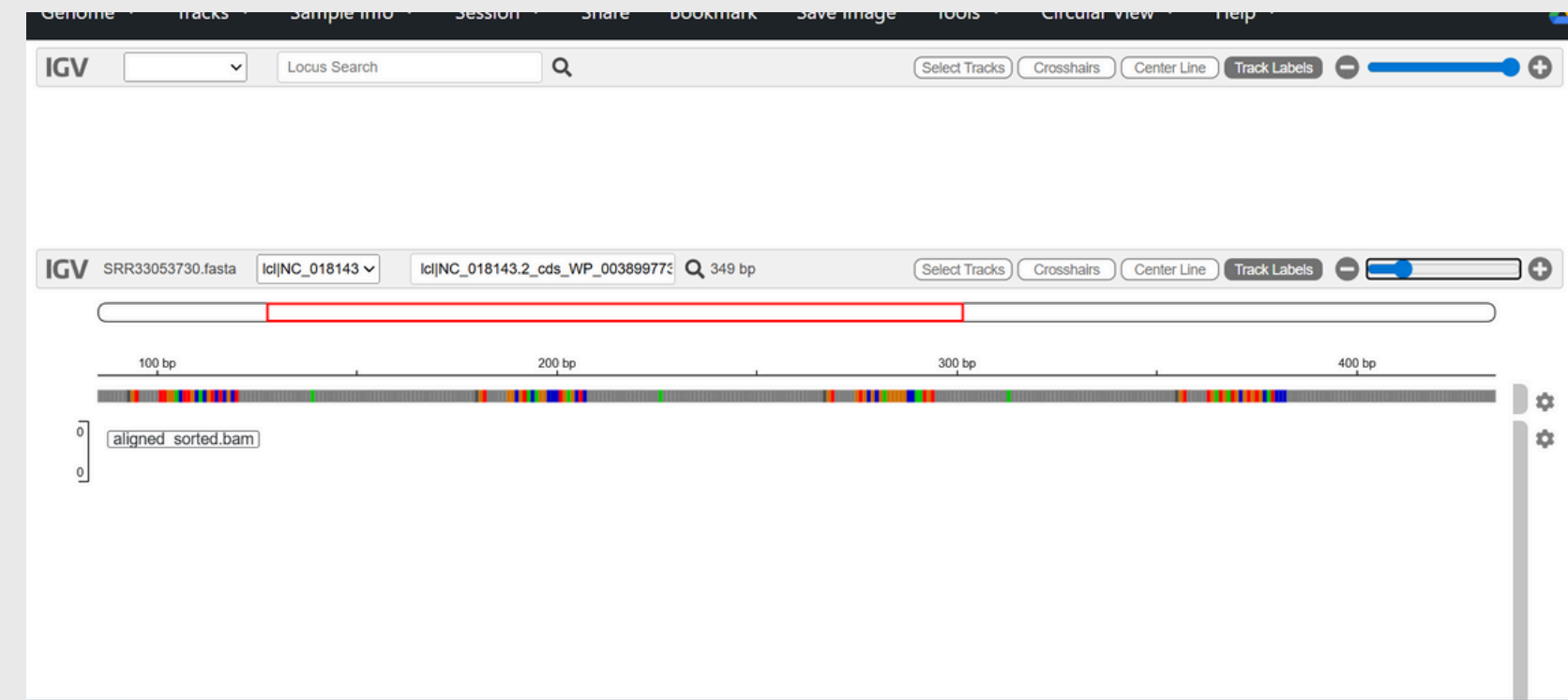
application/gzip
Started analysis of SRR33053730_trimmed.fastq.gz
Approx 5% complete for SRR33053730_trimmed.fastq.gz
Approx 10% complete for SRR33053730_trimmed.fastq.gz
Approx 15% complete for SRR33053730_trimmed.fastq.gz
Approx 20% complete for SRR33053730_trimmed.fastq.gz
Approx 25% complete for SRR33053730_trimmed.fastq.gz
Approx 30% complete for SRR33053730_trimmed.fastq.gz
Approx 35% complete for SRR33053730_trimmed.fastq.gz
Approx 40% complete for SRR33053730_trimmed.fastq.gz
Approx 45% complete for SRR33053730_trimmed.fastq.gz
Approx 50% complete for SRR33053730_trimmed.fastq.gz
Approx 55% complete for SRR33053730_trimmed.fastq.gz
Approx 60% complete for SRR33053730_trimmed.fastq.gz
Approx 65% complete for SRR33053730_trimmed.fastq.gz
Approx 70% complete for SRR33053730_trimmed.fastq.gz
Approx 75% complete for SRR33053730_trimmed.fastq.gz
Approx 80% complete for SRR33053730_trimmed.fastq.gz
Approx 85% complete for SRR33053730_trimmed.fastq.gz
Approx 90% complete for SRR33053730_trimmed.fastq.gz
Approx 95% complete for SRR33053730_trimmed.fastq.gz
Analysis complete for SRR33053730_trimmed.fastq.gz
(base) NGS $ gunzip GCF_000277735.2_ASM27773v2_cds_from_genomic.fna.gz
(base) NGS $ bwa index GCF_000277735.2_ASM27773v2_cds_from_genomic.fna
[bwa_index] Pack FASTA... 0.04 sec
[bwa_index] Construct BWT for the packed sequence...
[bwa_index] 0.89 seconds elapsed.
[bwa_index] Update BWT... 0.02 sec
```

```
[bwa_index] Pack forward-only FASTA... 0.02 sec
[bwa_index] Construct SA from BWT and Occ... 0.25 sec
[main] Version: 0.7.19-r1273
[main] CMD: bwa index GCF_000277735.2_ASM27773v2_cds_from_genomic.fna
[main] Real time: 1.242 sec; CPU: 1.244 sec
(base) NGS $ bwa mem GCF_000277735.2_ASM27773v2_cds_from_genomic.fna SRR33053730_trimmed.fastq.gz >
aligned.sam
[M::bwa_idx_load_from_disk] read 0 ALT contigs
[M::process] read 498440 sequences (1000001 bp)...
[M::process] read 498426 sequences (1000013 bp)...
[M::mem_process_seqs] Processed 498440 reads in 2.217 CPU sec, 1.985 real sec
[M::process] read 498444 sequences (1000027 bp)...
[M::mem_process_seqs] Processed 498426 reads in 2.321 CPU sec, 1.963 real sec
[M::process] read 498426 sequences (1000002 bp)...
[M::mem_process_seqs] Processed 498444 reads in 2.301 CPU sec, 1.955 real sec
[M::process] read 498446 sequences (1000031 bp)...
[M::mem_process_seqs] Processed 498426 reads in 2.337 CPU sec, 1.975 real sec
[M::process] read 498442 sequences (1000019 bp)...
[M::mem_process_seqs] Processed 498446 reads in 2.293 CPU sec, 1.936 real sec
[M::process] read 498426 sequences (1000036 bp)...
[M::mem_process_seqs] Processed 498442 reads in 2.334 CPU sec, 1.976 real sec
[M::process] read 486354 sequences (9757611 bp)...
[M::mem_process_seqs] Processed 498426 reads in 2.352 CPU sec, 1.991 real sec
[M::mem_process_seqs] Processed 486354 reads in 2.062 CPU sec, 1.953 real sec
[main] Version: 0.7.19-r1273
[main] CMD: bwa mem GCF_000277735.2_ASM27773v2_cds_from_genomic.fna SRR33053730_trimmed.fastq.gz
[main] Real time: 16.126 sec; CPU: 18.609 sec
(base) NGS $ samtools view -Sb aligned.sam >aligned.bam
(base) NGS $ samtools sort aligned.bam -o aligned_sorted.bam
(base) NGS $ samtools index aligned_sorted.bam
(base) NGS $ samtools faidx GCF_000277735.2_ASM27773v2_cds_from_genomic.fna
(base) NGS $
```

# STEPS FOR IGV ANALYSIS

## MYCOBACTERIUM TUBERCULOSIS H37RV

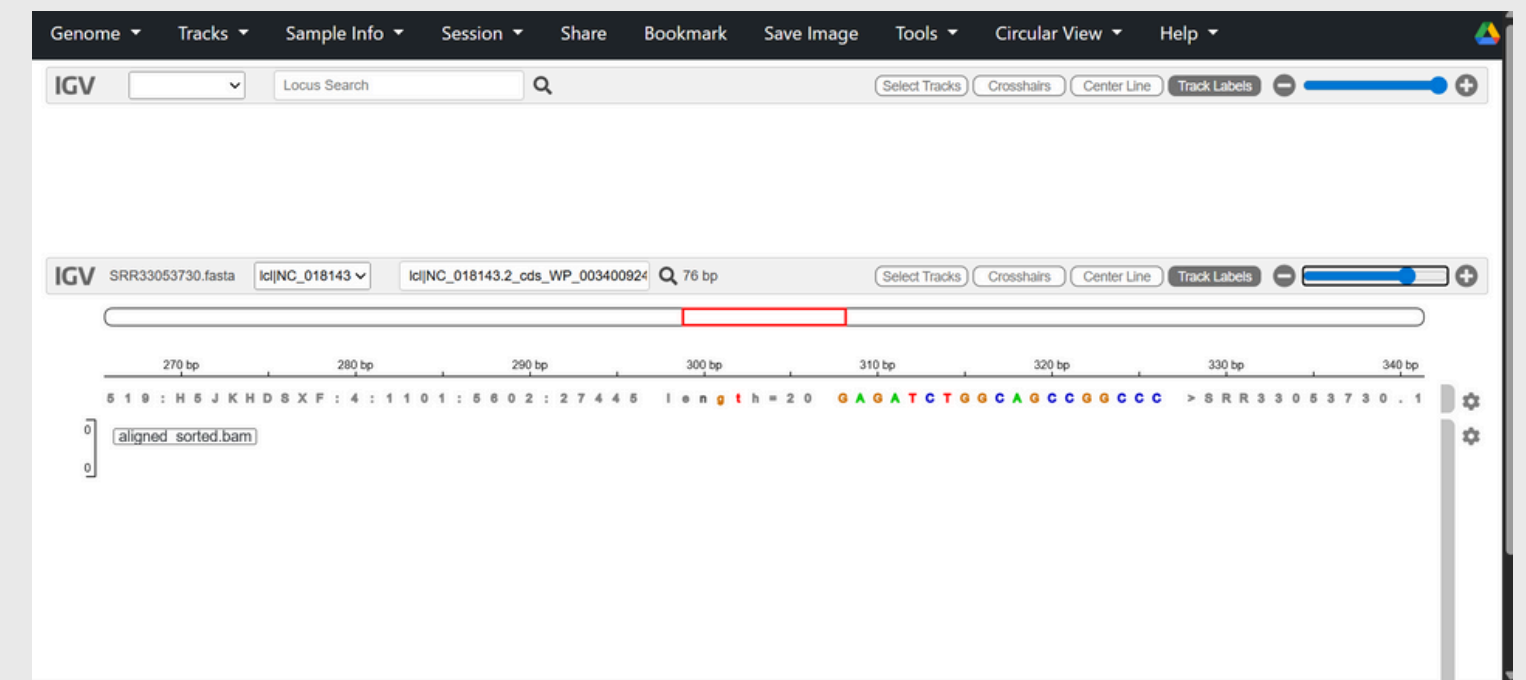
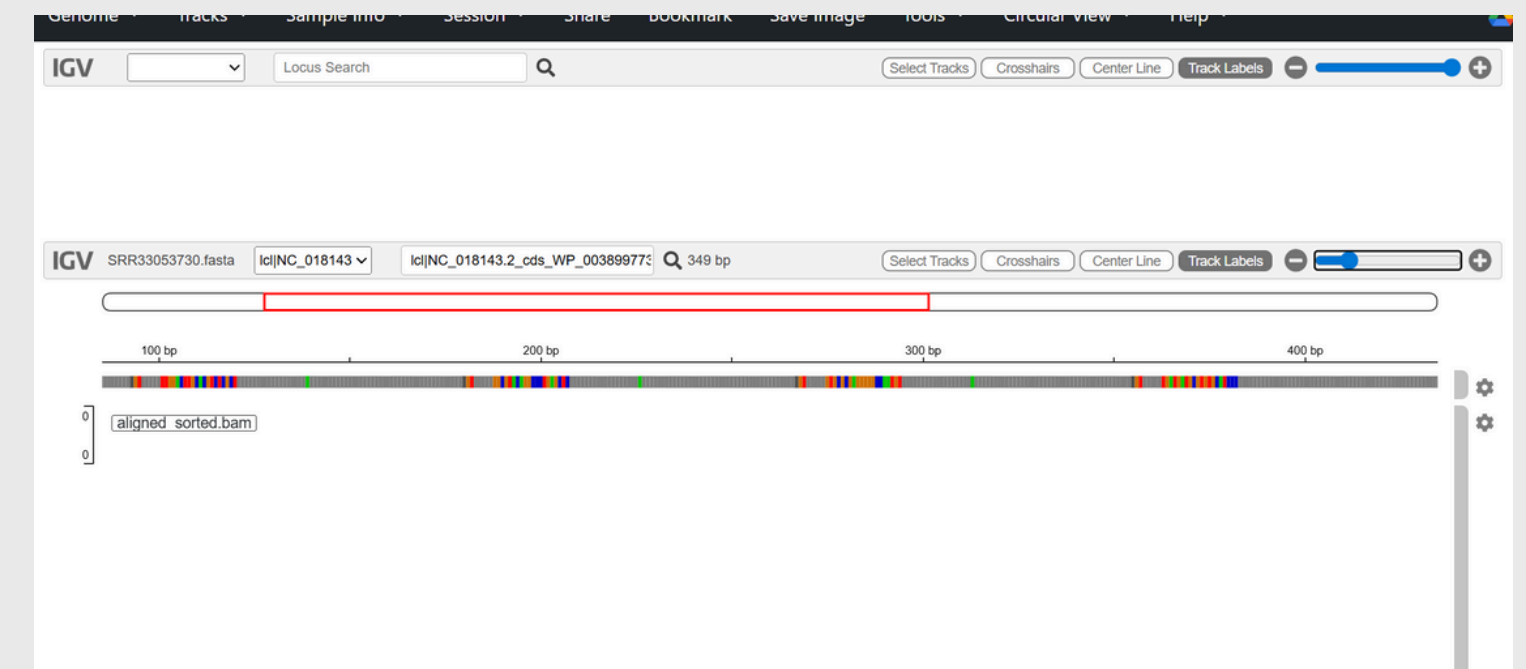
- Launch IGV
- Select Reference Genome
- Genomes → Load Genome from File
- Load BAM File
- File → Load from File...
  - Eg - Load: SRR33053730\_sorted.bam
- IGV will auto-detect and use SRR33053730\_sorted.bam.bai
- Navigate and Analyze in IGV
- Use the search bar to go to a gene or region





# IGV VISUALISATION

## MYCOBACTERIUM TUBERCULOSIS H37RV





# Q&A

## MYCOBACTERIUM TUBERCULOSIS H37RV

### What Changed After Trimming?

After trimming, the difference was pretty noticeable:

- The quality of the reads improved a lot – the ugly quality drop at the end was gone.
- The adapters were completely removed, which FASTQC confirmed.
- The reads were shorter overall, but the ones we kept were much cleaner.
- Duplication levels dropped a little too, though not completely – maybe due to PCR during library prep.
- Basically, after trimming, the data felt more trustworthy and ready for mapping.

### What We Saw in IGV?


Using IGV to visualize the reads was honestly one of the most satisfying parts. It's one thing to run alignment stats, but actually seeing the reads map to the genome made it real.

Most reads were nicely aligned, covering exons the way we'd expect.

In RNA-Seq data, we could see splicing patterns – split reads showing those little arcs across exons – which confirmed that alignment worked.

Some regions had gaps or lower coverage, which might reflect low expression or technical dropout.

And in variant views, we could spot some potential SNPs, which matched known positions.



**THANK YOU**

---