

Preprocessing reads: adapter trimming and quality filtering

Trimmomatic is a widely used tool for trimming adapter and PCR primer sequences from DNA and RNA-seq reads. It provides several different functions, including adapter clipping, quality clipping, and discarding reads below a certain quality or length threshold. Trimmomatic is a modular tool, allowing users to perform only the specific functions they require, which can save time and computational resources.

To use Trimmomatic, the first step is to determine which library preparation kit was used and whether the library was run in **paired-end** or **single-end mode**. This information is necessary to configure the trimming parameters appropriately.

Trimmomatic includes a variety of options and parameters that can be customized to suit your specific needs. Some commonly used options include:

1. **Adapter clipping:** This function removes adapter sequences from the reads, which can improve the accuracy of downstream analyses. Trimmomatic can automatically detect and remove adapter sequences based on the sequencing platform and library preparation kit used.
2. **Quality clipping:** This function removes low-quality bases from the reads, which can improve the accuracy of downstream analyses. Trimmomatic uses a sliding window approach to identify and remove bases with low quality scores.
3. **Discarding reads** below a certain quality or length threshold: This function removes reads that are below a certain length threshold, which can improve the accuracy of downstream analyses. Trimmomatic allows users to specify the minimum read length and quality score cut-offs.

Step options:

- ILLUMINACLIP:<fastaWithAdaptersEtc>:<seed mismatches>:<palindrome clip threshold>:<simple clip threshold>
 - fastaWithAdaptersEtc: specifies the path to a fasta file containing all the adapters, PCR sequences etc. The naming of the various sequences within this file determines how they are used. See below.
 - seedMismatches: specifies the maximum mismatch count which will still allow a full match to be performed
 - palindromeClipThreshold: specifies how accurate the match between the two 'adapter ligated' reads must be for PE palindrome read alignment.
 - simpleClipThreshold: specifies how accurate the match between any adapter etc. sequence must be against a read.
- SLIDINGWINDOW:<windowSize>:<requiredQuality>
 - windowSize: specifies the number of bases to average across
 - requiredQuality: specifies the average quality required.
- LEADING:<quality>
 - quality: Specifies the minimum quality required to keep a base.
- TRAILING:<quality>

- quality: Specifies the minimum quality required to keep a base.
- CROP:<length>
 - length: The number of bases to keep, from the start of the read.
- HEADCROP:<length>
 - length: The number of bases to remove from the start of the read.
- MINLEN:<length>
 - length: Specifies the minimum length of reads to be kept.

Trimmomatic is typically run as a command-line tool, with options and parameters specified in the command line.

```
cd
cd course
wget http://www.usadellab.org/cms/uploads/supplementary/Trimmomatic/Trimmomatic-0.39.zip
unzip Trimmomatic-0.39.zip
ls
```

you might have to install stuff !

For example, to trim adapter sequences and remove low-quality bases from single-end RNA-seq reads, keeping at least reads of length 70 and removing the first 5 bases from the beginning you might use the following command:

```
java -jar ./Trimmomatic-0.39/trimmomatic-0.39.jar SE SRR17655898.fastq.gz SRR17655898.trim.fastq.gz
ILLUMINACLIP:./Trimmomatic-0.39/adapters/TruSeq3-SE.fa:2:30:10 HEADCROP:5
SLIDINGWINDOW:4:15 MINLEN:70
```

Repeat this for the other 3 files accordingly

```
java -jar ./Trimmomatic-0.39/trimmomatic-0.39.jar SE SRR17655903.fastq.gz SRR17655903.trim.fastq.gz
ILLUMINACLIP:./Trimmomatic-0.39/adapters/TruSeq3-SE.fa:2:30:10 HEADCROP:5
SLIDINGWINDOW:4:15 MINLEN:70
```

```
java -jar ./Trimmomatic-0.39/trimmomatic-0.39.jar SE SRR17655906.fastq.gz SRR17655906.trim.fastq.gz
ILLUMINACLIP:./Trimmomatic-0.39/adapters/TruSeq3-SE.fa:2:30:10 HEADCROP:5
SLIDINGWINDOW:4:15 MINLEN:70
```

```
java -jar ./Trimmomatic-0.39/trimmomatic-0.39.jar SE SRR17655907.fastq.gz SRR17655907.trim.fastq.gz
ILLUMINACLIP:./Trimmomatic-0.39/adapters/TruSeq3-SE.fa:2:30:10 HEADCROP:5
SLIDINGWINDOW:4:15 MINLEN:70
```

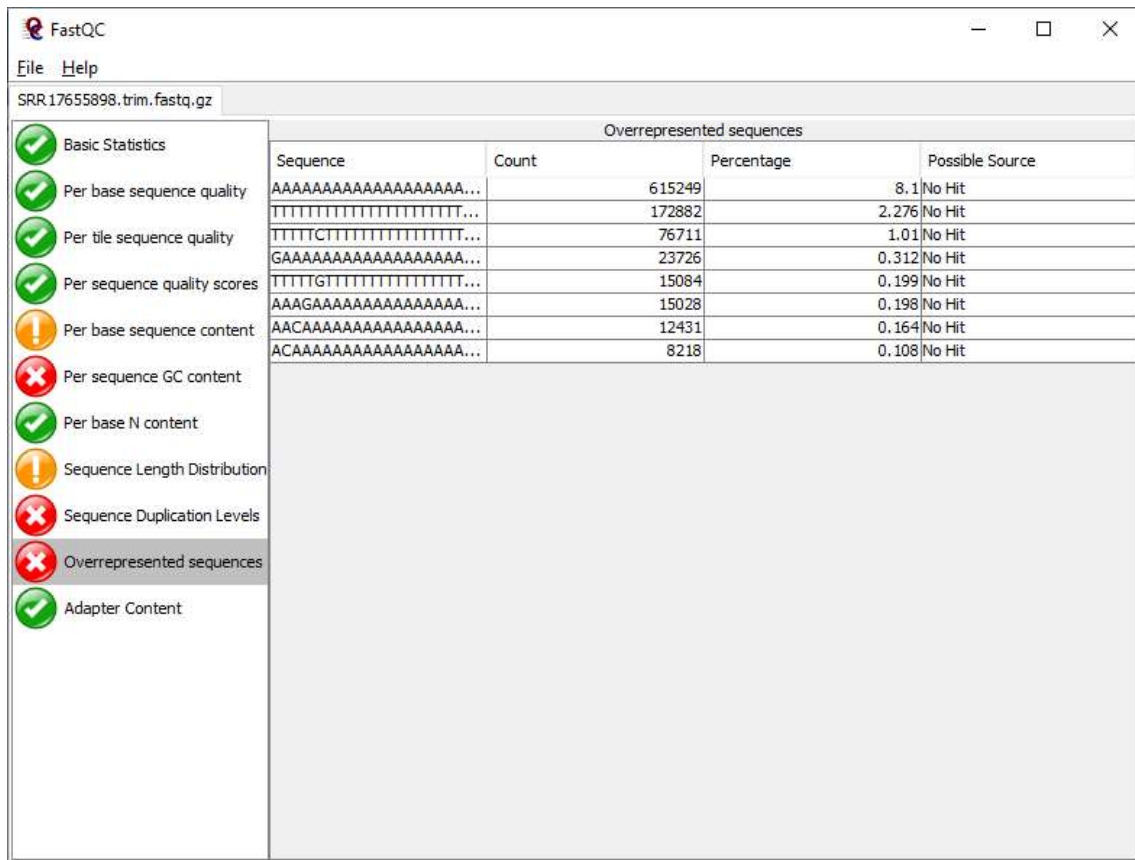
In theory you can name your files as you like But the next step will rely on the ending fastq.gz. So much for Unix flexibility

Always mind the output. It is good to keep this in a log file. Trimmomatic gives this in STDERR so we would need `2>trim.xxx.log` for piping.

```
(base) usadel@DESKTOP-N4USCVF:~/course$ java -jar ./Trimmomatic-0.39/trimmomatic-0.39.jar SE SRR17655898.fastq.gz SRR17655898.trim.gz ILLUMINACLIP:./Trimmomatic-0.39/adapters/TruSeq3-SE.fa:2:30:10 HEADCROP:5 SLIDINGWINDOW:4:15 MINLEN:70
TrimmomaticSE: Started with arguments:
SRR17655898.fastq.gz SRR17655898.trim.gz ILLUMINACLIP:./Trimmomatic-0.39/adapters/TruSeq3-SE.fa:2:30:10 HEADCROP:5 SLIDINGWINDOW:4:15 MINLEN:70
Automatically using 1 threads
Using Long Clipping Sequence: 'AGATCGGAAGAGCGTCGTGTAGGGAAAGAGTGT'
Using Long Clipping Sequence: 'AGATCGGAAGAGCACACGTCTGAACTCCAGTCAC'
ILLUMINACLIP: Using 0 prefix pairs, 2 forward/reverse sequences, 0 forward only sequences, 0 reverse only sequences
Quality encoding detected as phred33
Input Reads: 8658483 Surviving: 7595647 (87.81%) Dropped: 1054836 (12.19%)
TrimmomaticSE: Completed successfully
(base) usadel@DESKTOP-N4USCVF:~/course$
```

We can recheck our files in FASTQC

| Basic sequence stats | |
|-----------------------------------|---------------------------|
| Measure | Value |
| Filename | SRR17655898.trim.fastq.gz |
| File type | Conventional base calls |
| Encoding | Sanger / Illumina 1.9 |
| Total Sequences | 7595647 |
| Sequences flagged as poor quality | 0 |
| Sequence length | 70-96 |
| %GC | 35 |



FastQC window showing analysis results for SRR17655898.trim.fastq.gz. The 'Overrepresented sequences' section is expanded, showing a table of sequences that appear more frequently than expected.

| Overrepresented sequences | | | |
|---------------------------|--------|------------|-----------------|
| Sequence | Count | Percentage | Possible Source |
| AAAAAAAAAAAAAAAAAAAA... | 615249 | 8.1 | No Hit |
| TTTTTTTTTTTTTTTTTTTT... | 172882 | 2.276 | No Hit |
| TTTTTCTTTTTTTTTTTTTT... | 76711 | 1.01 | No Hit |
| GAAAAAAAAAAAAAAAAAAAA... | 23726 | 0.312 | No Hit |
| TTTTTGTTTTTTTTTTTTTT... | 15084 | 0.199 | No Hit |
| AAAGAAAAAAAAAAAAAAAA... | 15028 | 0.198 | No Hit |
| AACAAAAAAAAAAAAAAAAAA... | 12431 | 0.164 | No Hit |
| ACAAAAAAAAAAAAAAAAAAA... | 8218 | 0.108 | No Hit |

In this special case -public and easily accessible data- we can delete the original files to save some space

Analyzing data

We now map the reads against the genome for this we use hisat2

HISAT2

HISAT2 is a fast and “accurate” read mapper that aligns high-throughput sequencing reads to a reference genome.

One of the key features of HISAT2 is its use of a hierarchical indexing scheme based on the Burrows-Wheeler Transform (BWT), which enables it to rapidly search for potential alignment locations in the reference genome.

HISAT2 is a popular and efficient tool for aligning RNA-seq reads to a reference genome. The following steps provide a general overview of how to use HISAT2 for RNA-seq alignment:

1. Prepare a reference genome: HISAT2 requires a reference genome to align the reads to. The reference genome can be downloaded from a public repository such as NCBI or

- EBI or eventually just have been assembled by you. The reference genome needs to be indexed using HISAT2's indexing tool prior to alignment.
2. Prepare RNA-seq reads: RNA-seq reads should be preprocessed to remove adapter sequences and low-quality bases using tools like Trimmomatic. In any case the reads should be in a FastQ format.
 3. Align RNA-seq reads: To align RNA-seq reads to the reference genome using HISAT2, the following command can be used:

Hence, we must build an index for searching.

In the simplest case `hisat2-build genomefileinfasta INDEXNAME`

It makes sense to give the Index a meaningful name and document this very well

After that we can go ahead and search

- `-q`: Input file(s) are in FASTQ format
- `-x`: Path to the HISAT2 index files for the reference genome
- `-U`: Input file(s) contain unpaired reads
- `-1` and `-2`: Input file(s) contain paired-end reads, with `-1` specifying the file with the first mate and `-2` specifying the file with the second mate
- `-S`: Path to the output SAM file
- `--rna-strandness`: Specifies the strand-specificity of the RNA-seq data. Options are `FR`, `RF`, and `FF`.
- `--dta`: for gene definition
- `--threads` or `-p`: Number of threads to use for alignment (default is 1)

```

cd course
LINUX (UBUNTU on WINDOWS)
wget https://cloud.biohpc.swmed.edu/index.php/s/oTtGWbWjajsQ2Ho/download
unzip download
wget
https://ftp.ncbi.nlm.nih.gov/genomes/all/GCA/022/817/965/GCA\_022817965.1\_SlydLA2951\_v2.0/GCA\_022817965.1\_SlydLA2951\_v2.0\_genomic.fna.gz

ONLY FOR MAC
curl https://cloud.biohpc.swmed.edu/index.php/s/zMgEtnF6LjnjFrr/download --output hisatosx.zip
unzip hisatosx.zip
curl
https://ftp.ncbi.nlm.nih.gov/genomes/all/GCA/022/817/965/GCA\_022817965.1\_SlydLA2951\_v2.0/GCA\_022817965.1\_SlydLA2951\_v2.0\_genomic.fna.gz --output GCA_022817965.1_SlydLA2951_v2.0_genomic.fna.gz

BOTH
gunzip GCA_022817965.1_SlydLA2951_v2.0_genomic.fna.gz
./hisat2-2.2.1/hisat2-build GCA_022817965.1_SlydLA2951_v2.0_genomic.fna.gz PERSI

```

Now we run Hisat2 in unpaired (single ended) mode as we only have single ended reads

```

./hisat2-2.2.1/hisat2 -x PERSI -U SRR17655898.trim.fastq.gz -p 4 -S SRR17655898.sam
./hisat2-2.2.1/hisat2 -x PERSI -U SRR17655903.trim.fastq.gz -p 4 -S SRR17655903.sam
./hisat2-2.2.1/hisat2 -x PERSI -U SRR17655906.trim.fastq.gz -p 4 -S SRR17655906.sam
./hisat2-2.2.1/hisat2 -x PERSI -U SRR17655907.trim.fastq.gz -p 4 -S SRR17655907.sam

```

-x gives the reference we had just built

-U is for unpaired reads

-p is for multiple threads (if you have that)

-S is for sam output

Once again we get very valuable information that we should keep by using `2>` or by noting it down

```
(ERR): hisat2-align exited with value 1
(base) usadel@DESKTOP-M4USCVF:~/course$ ./hisat2-2.2.1/hisat2 -x PERSI -U SRR17655898.trim.fastq.gz -p 4 -S SRR17655898.sam
7595647 reads; of these:
  7595647 (100.00%) were unpaired; of these:
    1432272 (18.86%) aligned 0 times
    3304374 (43.50%) aligned exactly 1 time
    2859001 (37.64%) aligned >1 times
81.14% overall alignment rate
(base) usadel@DESKTOP-M4USCVF:~/course$
```

Samtools: Getting the pocket knife

Ubuntu: Simple solutions but it will install an old version if it works

```
sudo apt install samtools
#check if it works
samtools
```

if it works in the following just use samtools instead of ./samtools-1.9/samtools

normally compile everything

```
sudo apt-get update
sudo apt-get install gcc
sudo apt-get install make
sudo apt-get install libbz2-dev
sudo apt-get install zlib1g-dev
sudo apt-get install libncurses5-dev
sudo apt-get install libncursesw5-dev
sudo apt-get install liblzma-dev
sudo apt-get install libncurses5-dev libncursesw5-dev
```

```
cd
cd course
wget https://github.com/samtools/htslib/releases/download/1.9/htslib-1.9.tar.bz2
tar -xjf htslib-1.9.tar.bz2
cd htslib-1.9
make
```



```
cd ..
wget https://github.com/samtools/samtools/releases/download/1.9/samtools-1.9.tar.bz2
tar -xjf samtools-1.9.tar.bz2
cd samtools-1.9
make
```

We just sort convert to bam and then index the files

If you are short in hard disk space convert a file then delete the sam file. Typically one would redirect hisat's output directly to bam file using "|"

```
./samtools-1.9/samtools sort SRR17655898.sam -o SRR17655898.bam
./samtools-1.9/samtools index SRR17655898.bam

./samtools-1.9/samtools sort SRR17655906.sam -o SRR17655906.bam
./samtools-1.9/samtools index SRR17655906.bam

./samtools-1.9/samtools sort SRR17655907.sam -o SRR17655907.bam
./samtools-1.9/samtools index SRR17655907.bam

./samtools-1.9/samtools sort SRR17655903.sam -o SRR17655903.bam
./samtools-1.9/samtools index SRR17655903.bam

#let's have a look how file sizes change of all that ends in "am"
ls -alh *am

#not bad so we can delete all SAM files now
#!!!!!! Careful only do this is if you are sure you have all bam files
rm *sam
```

IGV Displaying and analyzing data

IGV

Get IGV for Mac or Windows or Linux <https://software.broadinstitute.org/software/igv/download>

You might need JAVA to run e.g. from here

<https://adoptopenjdk.net/releases.html>

The Integrative Genomics Viewer (IGV) is a powerful tool for visualizing and exploring genomic data. It is widely used by researchers in genomics, genetics, and bioinformatics to visualize and analyze data from a variety of sequencing experiments, including RNA-seq, ChIP-seq, and DNA sequencing.

IGV allows users to view genomic data in a graphical interface, making it easy to identify patterns and anomalies in the data. Users can zoom in and out of the genome, navigate to specific genomic regions, and view a range of data types, including aligned reads, variant calls, and genome annotations.

One of the key features of IGV is its ability to visualize sequencing data in real-time. This means that as the user navigates through the genome, IGV dynamically loads and displays the relevant data, providing a seamless and responsive user experience.

IGV also provides a range of analysis tools and plugins that allow users to explore and analyze their data in greater detail. For example, users can use the peak-calling tool to identify genomic regions of interest, or the differential expression tool to identify genes that are differentially expressed between different samples.

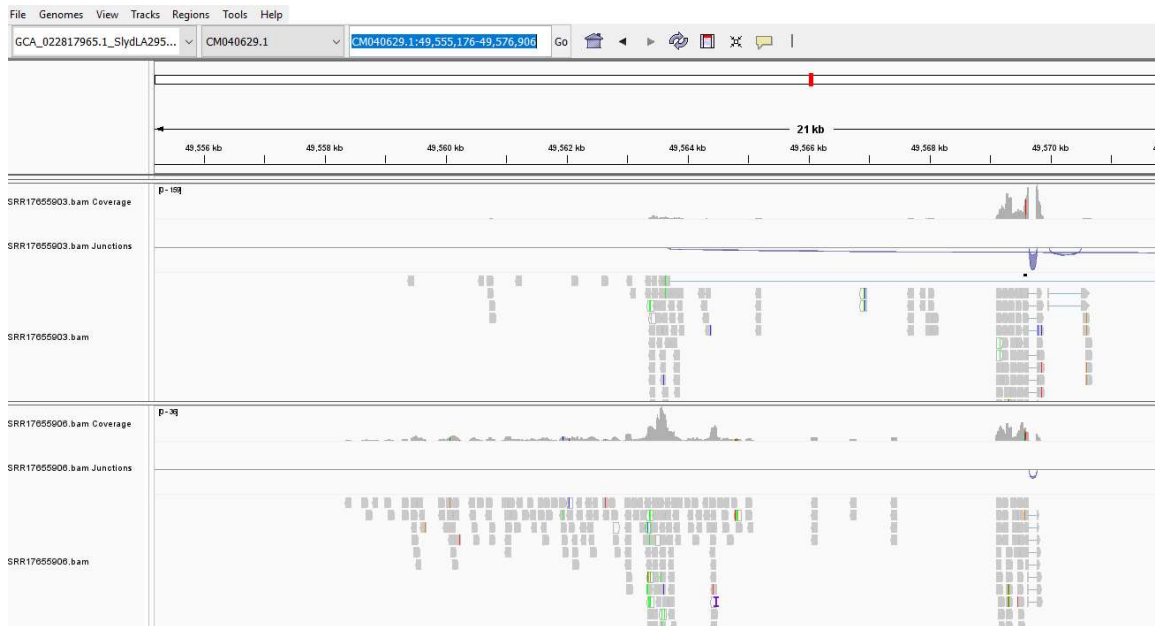
To use IGV, users first need to download and install the software on their computer. They can then load their genomic data into the program and begin exploring and analyzing the data using the graphical interface and analysis tools.

1. Load data: To load the genome data into IGV, click "Genomes" in the top menu and select "Load Genome from File" or "Load from URL". This will open a dialog box where you can select your data file. You might have to rename the genome file to XXX.fasta (where XXX is any name you like : no spaces no special characters)
2. Next load your indexed bam files Click File=>load from file Load at least ...03.bam and ..06.bam
3. Navigate the genome: Once your data is loaded, you can navigate the genome using the graphical interface. You can zoom in and out by scrolling with your mouse or trackpad, or by using the zoom slider in the top left corner of the window. You can also use the "Go To" box to navigate to a specific genomic location.
4. View data: To view your data, click on the "Tracks" menu and select the tracks you want to view. Aligned reads, annotations, and other data types can be added to the view. You can adjust the display settings of the tracks, such as color and height, by right-clicking on the track and selecting "Configure".

Move around in the data check location

CM040629.1:49,555,176-49,576,906

(you can paste this into the box next to GO)



What is happening here?

Look very carefully!

But alas we are missing something. Where are the genes?