Samtools

Learning outcomes

After having completed this chapter you will be able to:

- Use `samtools flagstat` to get general statistics on the flags stored in a sam/bam file
- Use `samtools view` to:
  - compress a sam file into a bam file
  - filter on sam flags
  - count alignments
  - filter out a region
- Use `samtools sort` to sort an alignment file based on coordinate
- Use `samtools index` to create an index of a sorted sam/bam file
- Use the pipe (`|`) symbol to pipe alignments directly to `samtools` to perform sorting and filtering

Material

- `samtools` documentation
- Explain sam flags tool

Exercises

Alignment statistics

Exercise: Write the statistics of the E. coli alignment to file called `SRR519926.sam.stats` by using `samtools flagstat`. Find the documentation here. Anything that draws your attention?

Answer

Code:

```
cd ~/project/results/alignments/
samtools flagstat SRR519926.sam > SRR519926.sam.stats
```

resulting in:

```
624724 + 0 in total (QC-passed reads + QC-failed reads)
624724 + 0 primary
0 + 0 secondary
0 + 0 supplementary
0 + 0 duplicates
0 + 0 primary duplicates
621624 + 0 mapped (99.50% : N/A)
621624 + 0 primary mapped (99.50% : N/A)
624724 + 0 paired in sequencing
312362 + 0 read1
312362 + 0 read2
300442 + 0 properly paired (48.09% : N/A)
619200 + 0 with itself and mate mapped
2424 + 0 singletons (0.39% : N/A)
0 + 0 with mate mapped to a different chr
0 + 0 with mate mapped to a different chr (mapQ>=5)
```

Of the reads, 47.87% is properly paired. The rest isn't. Proper pairing is quite hard to interpret. It usually means that the 0x2 flag (each segment properly aligned according to the aligner) is false. In this case it means that the insert size is high for a lot of sequences. That is because the insert size distribution is very wide. You can find info on insert size distribution like this:

```
samtools stats SRR519926.sam | grep ^SN | cut -f 2,3
```

Now look at `insert size average` and `insert size standard deviation`. You can see the standard deviation is higher than the average, suggesting a wide distribution.

Compression, sorting and indexing

The command `samtools view` is very versatile. It takes an alignment file and writes a filtered or processed alignment to the output. You can for example use it to compress your SAM file into a BAM file. Let's start with that.

Exercise: Create a script called `08_compress_sort.sh`. Add a `samtools view` command to compress our SAM file into a BAM file and include the header in the output. For this, use the `-b` and `-h` options. Find the required documentation here. How much was the disk space reduced by compressing the file?

Tip: Samtools writes to stdout

By default, samtools writes it's output to stdout. This means that you need to redirect your output to a file with `>` or use the the output option `-o`.

Answer

08_compress_sort.sh

```
#!/usr/bin/env bash

cd ~/project/results/alignments

samtools view -bh SRR519926.sam > SRR519926.bam
```

By using `ls -lh`, you can find out that `SRR519926.sam` has a size of 264 Mb, while `SRR519926.bam` is only 77 Mb.

To look up specific alignments, it is convenient to have your alignment file indexed. An indexing can be compared to a kind of 'phonebook' of your sequence alignment file. Indexing is done with `samtools` as well, but it first needs to be sorted on coordinate (i.e. the alignment location). You can do it like this:

```
samtools sort SRR519926.bam > SRR519926.sorted.bam
samtools index SRR519926.sorted.bam
```

Exercise: Add these lines to `08_compress_sort.sh`, and re-run te script in order to generate the sorted bam file. After that checkout the headers of the unsorted bam file (`SRR519926.bam`) and the sorted bam file (`SRR519926.sorted.bam`) with `samtools view -H`. What are the differences?

Answer

Your script should like like this:

08_compress_sort.sh

```
#!/usr/bin/env bash

cd ~/project/results/alignments

samtools view -bh SRR519926.sam > SRR519926.bam
samtools sort SRR519926.bam > SRR519926.sorted.bam
samtools index SRR519926.sorted.bam
```

`samtools view -H SRR519926.bam` returns:

```
@HD     VN:1.0  SO:unsorted
@SQ     SN:U00096.3     LN:4641652
@PG     ID:bowtie2      PN:bowtie2      VN:2.4.2
CL:"/opt/conda/envs/ngs-tools/bin/bowtie2-align-s --wrapper basic-0 -x
/config/project/ref_genome//ecoli-strK12-MG1655.fasta -1
/config/project/trimmed_data/trimmed_SRR519926_1.fastq -2
/config/project/trimmed_data/trimmed_SRR519926_2.fastq"
@PG     ID:samtools     PN:samtools     PP:bowtie2      VN:1.12 CL:samtools
view -bh SRR519926.sam
@PG     ID:samtools.1   PN:samtools     PP:samtools     VN:1.12 CL:samtools
view -H SRR519926.bam
```

And `samtools view -H SRR519926.sorted.bam` returns:

```
@HD     VN:1.0  SO:coordinate
@SQ     SN:U00096.3     LN:4641652
@PG     ID:bowtie2      PN:bowtie2      VN:2.4.2
CL:"/opt/conda/envs/ngs-tools/bin/bowtie2-align-s --wrapper basic-0 -x
/config/project/ref_genome//ecoli-strK12-MG1655.fasta -1
/config/project/trimmed_data/trimmed_SRR519926_1.fastq -2
/config/project/trimmed_data/trimmed_SRR519926_2.fastq"
@PG     ID:samtools     PN:samtools     PP:bowtie2      VN:1.12 CL:samtools
view -bh SRR519926.sam
@PG     ID:samtools.1   PN:samtools     PP:samtools     VN:1.12 CL:samtools
sort SRR519926.bam
@PG     ID:samtools.2   PN:samtools     PP:samtools.1   VN:1.12 CL:samtools
view -H SRR519926.sorted.bam
```

There are two main differences:

- The `SO` tag at `@HD` type code has changed from `unsorted` to `coordinate`.
- A line with the `@PG` type code for the sorting was added.

Note that the command to view the header (`samtools -H`) is also added to the header for both runs.

Filtering

With `samtools view` you can easily filter your alignment file based on flags. One thing that might be sensible to do at some point is to filter out unmapped reads.

Exercise: Check out the flag that you would need to filter for mapped reads. It's at page 7 of the SAM documentation.

Answer

You will need the 0x4 flag.

Filtering against unmapped reads (leaving only mapped reads) with `samtools view` would look like this:

```
samtools view -bh -F 0x4 SRR519926.sorted.bam > SRR519926.sorted.mapped.bam
```

or:

```
samtools view -bh -F 4 SRR519926.sorted.bam > SRR519926.sorted.mapped.bam
```

Exercise: Generate a script called `09_extract_unmapped.sh` to get only the unmapped reads (so the opposite of the example). How many reads are in there? Is that the same as what we expect based on the output of `samtools flagstat`?

Tip

Check out the `-f` and `-c` options of `samtools view`

Answer

Your script `09_extract_unmapped.sh` should look like this:

09_extract_unmapped.sh

```
#!/usr/bin/env bash

cd ~/project/results/alignments

samtools view -bh -f 0x4 SRR519926.sorted.bam > SRR519926.sorted.unmapped.bam
```

Counting like this:

```
samtools view -c SRR519926.sorted.unmapped.bam
```

This should correspond to the output of `samtools flagstat` (624724 - 621624 = 3100)

`samtools view` also enables you to filter alignments in a specific region. This can be convenient if you don't want to work with huge alignment files and if you're only interested in alignments in a particular region. Region filtering only works for sorted and indexed alignment files.

Exercise: Generate a script called `10_extract_region.sh` to filter our sorted and indexed BAM file for the region between 2000 and 2500 kb, and output it as a BAM file with a header.

Tip: Specifying a region

Our E. coli genome has only one chromosome, because only one line starts with `>` in the fasta file

```
cd ~/project/ref_genome
grep ">" ecoli-strK12-MG1655.fasta
```

gives:

```
>U00096.3 Escherichia coli str. K-12 substr. MG1655, complete genome
```

The part after the first space in the title is cut off for the alignment reference. So the code for specifying a region would be: `U00096.3:START-END`

Answer

10_extract_region.sh

```
#!/usr/bin/env bash

cd ~/project/results/alignments

samtools view -bh \
SRR519926.sorted.bam \
U00096.3:2000000-2500000 \
> SRR519926.sorted.region.bam
```

Redirection

Samtools is easy to use in a pipe. In this case you can replace the input file with a `-`. For example, you can sort and compress the output of your alignment software in a pipe like this:

```
my_alignment_command \
| samtools sort - \
| samtools view -bh - \
> alignment.bam
```

The use of `-`

In the modern versions of samtools, the use of `-` is not needed for most cases, so without an input file it reads from stdin. However, if you're not sure, it's better to be safe than sorry.

Exercise: Write a script called `11_align_sort.sh` that maps the reads with bowtie2 (see chapter 2 of read alignment), sorts them, and outputs them as a BAM file with a header.

Answer

11_align_sort.sh

```
#!/usr/bin/env bash

TRIMMED_DIR=~/project/results/trimmed
REFERENCE_DIR=~/project/ref_genome
ALIGNED_DIR=~/project/results/alignments

bowtie2 \
-x $REFERENCE_DIR/ecoli-strK12-MG1655.fasta \
-1 $TRIMMED_DIR/trimmed_SRR519926_1.fastq \
-2 $TRIMMED_DIR/trimmed_SRR519926_2.fastq \
2> $ALIGNED_DIR/bowtie2_SRR519926.log \
| samtools sort - \
| samtools view -bh - \
> $ALIGNED_DIR/SRR519926.sorted.mapped.frompipe.bam
```

Redirecting `stderr`

Notice the line starting with `2>`. This redirects standard error to a file: `$ALIGNED_DIR/bowtie2_SRR519926.log`. This file now contains the bowtie2 logs, that can later be re-read or used in e.g. `multiqc`.

QC summary

The software MultiQC is great for creating summaries out of log files and reports from many different bioinformatic tools (including `fastqc`, `fastp`, `samtools` and `bowtie2`). You can specify a directory that contains any log files, and it will automatically search it for you.

Exercise: Run the command `multiqc .` in `~/project` and checkout the generated report.