Group work

The last part of this course will consist of project-based-learning. This means that you will work in groups on a single question. We will split up into groups of five people.

If working with Docker

If you are working with Docker, I assume you are working independently and therefore can not work in a group. However, you can test your skills with these real biological datasets. Realize that the datasets and calculations are (much) bigger compared to the exercises, so check if your computer is up for it. You'll probably need around 4 cores, 16G of RAM and 50G of harddisk.

If online

If the course takes place online, we will use break-out rooms to communicate within groups. Please stay in the break-out room during the day, also if you are working individually.

Material

Download the presentation

Roles & organisation

Project based learning is about learning by doing, but also about *peer instruction*. This means that you will be both a learner and a teacher. There will be differences in levels among participants, but because of that, some will learn efficiently from people that have just learned, and others will teach and increase their understanding.

Each project has **tasks** and **questions**. By performing the tasks, you should be able to answer the questions. You should consider the tasks and questions as a guidance. If interesting questions pop up during the project, you are **encouraged** to work on those. Also, you don't have to perform all the tasks and answer all the questions.

In the afternoon of day 1, you will start on the project. On day 3, you can work on the project in the morning and in the first part of the afternoon. We will conclude the projects with a **10-minute presentation** of each group.

Working directories

Each group has access to a shared working directory. It is mounted in the root directory (`/`). You can add this directory to your working space by clicking: **File > Add Folder to Workspace…**. Then, type the path to your group directory: `/group_work/groupX` (where `X` is your group number).

Project 1: Variant analysis of human data

Aim: Find variants on chromosome 20 from three samples

In this project you will be working with Illumina reads from three samples: a father, mother and a child. You will perform quality control, align the reads, mark duplicates, detect variants and visualize them.

You can get the data by running these commands:

```
wget https://ngs-introduction-training.s3.eu-central-
1.amazonaws.com/project1.tar.gz
tar -xvf project1.tar.gz
rm project1.tar.gz
```

Tasks

> **Important!**
>
> **Stick to the principles for reproducible analysis** described here

- Download the required data
- Do a QC on the data with `fastqc`
- Trim adapters and low quality bases with `fastp`. Make sure to include the option `--detect_adapter_for_pe`. To prevent overwriting `fastp.html`, specify a report filename for each sample with the option `--html`.
- After trimming the adapters, run `fastqc` again to see whether all adapters are gone.
- Create an index for bowtie2. At the same time create a fasta index (`.fai` file) with `samtools faidx`.
- Check which options to use, and align with `bowtie2`. At the same time add readgroups to the aligned reads (see hints below). Make sure you end up with an indexed and sorted bam file.
- Mark duplicates on the individual bam files with `gatk MarkDuplicates` (see hints below).

- Merge the three bam files with `samtools merge`. Index the bam file afterwards.
- Run `freebayes` to call variants. Only call variants on the region `chr20:10018000-10220000` by specifying the `-r` option.
- Load your alignments together with the vcf containing the variants in IGV. Check out e.g. `chr20:10,026,397-10,026,638`.
- Run `multiqc` to get an overall quality report.

Questions

- Have a look at the quality of the reads. Are there any adapters in there? Did adapter trimming change that? How is the base quality? Could you improve that?
- How many duplicates were in the different samples (hint: use `samtools flagstat`)? Why is it important to remove them for variant analysis?
- Why did you add read groups to the bam files? Where is this information added in the bam file?
- Are there variants that look spurious? What could be the cause of that? What information in the vcf can you use to evaluate variant quality?
- There are two high quality variants in `chr20:10,026,397-10,026,638`. What are the genotypes of the three samples according to freebayes? Is this according to what you see in the alignments? If the alternative alleles are present in the same individual, are they in phase or in repulsion? Note: you can also load vcf files in IGV.

Hints

You can add readgroups to the alignment file with `bowtie2` with the options `--rg-id` and `--rg`, e.g. (`$SAMPLE` is a variable containing a sample identifier):

```
bowtie2 \
-x ref.fa \
-1 r1.fastq.gz \
-2 r2.fastq.gz \
--rg-id $SAMPLE \
--rg SM:$SAMPLE \
```

To run `gatk MarkDuplicates` you will only need to specify `--INPUT` and `--OUTPUT`, e.g.:

```
gatk MarkDuplicates \
--INPUT sample.bam \
--OUTPUT sample.md.bam \
--METRICS_FILE sample.metrics.txt
```

Project 2: Long-read genome sequencing

Aim: Align long reads from RNA-seq data to a reference genome.

In this project, you will be working with data from:

> Padilla, Juan-Carlos A., Seda Barutcu, Ludovic Malet, Gabrielle Deschamps-Francoeur, Virginie Calderon, Eunjeong Kwon, and Eric Lécuyer. “Profiling the Polyadenylated Transcriptome of Extracellular Vesicles with Long-Read Nanopore Sequencing.” BMC Genomics 24, no. 1 (September 22, 2023): 564. https://doi.org/10.1186/s12864-023-09552-6.

The authors used RNA sequencing with Oxford Nanopore Technology of both extracellular vesicles and whole cells from cell culture. For this project, we will work with two samples of this study, `EV_2` (extracellular vesicle) and `Cell_2` (whole cell). Download and unpack the data files.

Download the human reference genome like this:

```
wget https://ngs-introduction-training.s3.eu-central-
1.amazonaws.com/project2.tar.gz
tar -xvf project2.tar.gz
rm project2.tar.gz
```

You can find the fastq files in the `reads` folder and the reference genome and its annotation in the `reference` folder. To reduce computational times we work with a subset of the data on a subset of the genome (chromosome 5 and X).

Tasks

> ⚠ **Important!**
>
> **Stick to the principles for reproducible analysis** described here

- Perform QC with `fastqc`
- Perform QC with `NanoPlot`
- Align with `minimap2` with default parameters
- Figure how you should set parameter `-x`
- Evaluate the alignment quality (e.g. alignment rates, mapping quality)
- Compare the two different samples in read quality, alignment rates, depth, etc.
- Check out the alignments in IGV. Check out e.g. `ELOVL5`.

Questions

- Have a look at the quality report. What are the average read lengths? Is that expected?
- What is the average read quality? What kind of accuracy would you expect?
- Note any differences between `fastqc` and `NanoPlot`? How is that compared to the publication?
- Check out the option `-x` of `minimap2`. Are the defaults appropriate?
- You might consider using `-x map-ont` or `-x splice`. Do you see differences in the alignment in e.g. IGV?
- How are spliced alignments stored in the SAM file with the different settings of `-x`?
- How deep is the gene `ELOVL5` sequenced in both samples?
- Do you already see evidence for splice variants in the alignments?

Accuracy from quality scores

Find the equation to calculate error probability from quality score on Wikipedia.

Comparing `fastqc` and `Nanoplot`

For comparing `fastqc` and `NanoPlot`, check out this blog of the author of NanoPlot, and this thread.

Running `minimap2`

Here's an example command for `minimap2`:

```
minimap2 \
-a \
-x [PARAMETER] \
[REFERENCE].fa \
[FASTQFILE].fastq.gz \
| samtools sort \
| samtools view -bh > [OUTPUT].bam
```

Project 3: Short-read RNA-seq of mice.

Aim: Generate a count matrix to estimate differential gene expression.

In this project you will be working with data from:

Singhania A, Graham CM, Gabryšová L, Moreira-Teixeira L, Stavropoulos E, Pitt JM, et al (2019). *Transcriptional profiling unveils type I and II interferon networks in blood and tissues across diseases*. Nat Commun. 10:1–21. https://doi.org/10.1038/s41467-019-10601-6

Here's the BioProject page. Since the mouse genome is rather large, we have prepared reads for you that originate from chromosome 5. Use those for the project. Download them like this:

```
wget https://ngs-introduction-training.s3.eu-central-
1.amazonaws.com/project3.tar.gz
tar -xvf project3.tar.gz
rm project3.tar.gz
```

Tasks

> ⚠ **Important!**
>
> **Stick to the principles for reproducible analysis** described here

- Download the tar file, and find out what's in the data folder
- Do a QC on the fastq files with `fastqc`
- Trim adapters and low quality bases with `fastp`
- After trimming the adapters, run `fastqc` again to see whether all adapters are gone.
- Check which options to use, and align with `hisat2`
- Evaluate the alignment quality (e.g. alignment rates, mapping quality)
- Have a look at the alignments in IGV, e.g. check out `Sparcl1`. For this, you can use the built-in genome (*Mouse (mm10)*). Do you see any evidence for differential splicing?
- Run `featureCounts` on both alignments. Have a look at the option `-Q`. For further suggestions, see the hints below.
- Compare the count matrices in `R` (find a script to get started here; Rstudio server is running on the same machine. Approach it with your credentials and username `rstudio`)

Questions

- Check the description at the SRA sample page. What kind of sample is this?
- How does the quality of the reads look? Anything special about the overrepresented sequences? (Hint: blast some overrepresented sequences, and see what they are)
- Did trimming improve the QC results? What could be the cause of the warnings/errors in the `fastqc` reports?
- What are the alignment rates?

- How are spliced alignments stored in the SAM file?
- Are there any differences between the treatments in the percentage of assigned alignments by `featureCounts`? What is the cause of this?
- Can you find any genes that seem to be differentially expressed?
- What is the effect of setting the option `-Q` in `featureCounts`?

Hints

We are now doing computations on a full genome, with full transcriptomic data. This is quite a bit more than we have used during the exercises. Therefore, computations take longer. However, most tools support parallel processing, in which you can specify how many cores you want to use to run in parallel. Your environment contains **four** cores, so this is also the maximum number of processes you can specify. Below you can find the options used in each command to specify multi-core processing.

command	option
`bowtie2-build`	`--threads`
`hisat2-build`	`--threads`
`fastqc`	`--threads`
`cutadapt`	`--cores`
`bowtie2`	`--threads`
`hisat2`	`--threads`
`featureCounts`	`-T`

Here's some example code for `hisat2` and `featureCounts`. Everything in between `<>` should be replaced with specific arguments.

Here's an example for `hisat2`:

```
hisat2-build <reference_sequence_fasta> <index_basename>

hisat2 \
-x <index_basename> \
-1 <foward_reads.fastq.gz> \
-2 <reverse_reads.fastq.gz> \
```

```
-p <threads> \
| samtools sort \
| samtools view -bh \
> <alignment_file.bam>
```

Example code `featureCounts`:

```
featureCounts \
-p \
-T 2 \
-a <annotations.gtf> \
-o <output.counts.txt> \
*.bam
```