NGS - quality control, alignment, visualisation

Reproducible computational research

Why reproducible?

1. For yourself!
   a. Adjusting your analysis
   b. Sharing your analysis
   c. Find out what the heck you did > 2 weeks ago
2. Because the academic community requires it..
   1. Many journals require accompanied code
   2. Proposals often require a data management plan

5 simple rules to get started

Rule 1 ✓

Execute the commands from a script in order to be able to trace back your steps

- All output **files** and **directories** created from within a script
- **Adjusting** your analysis becomes possible
- Makes your analysis **portable**. It can be run:
  - On a different computer
  - By your colleague

Rule 2 ✓

Number scripts based on their order of execution (e.g. `01_download_reads.sh`)

- Easily trace the order of execution
- Seperates **main** scripts from **secondary** scripts (i.e. scripts called by another script)

```
01_download_reads.sh
02_run_fastqc.sh
03_trim_reads.sh
04_run_fastqc_trimmed.sh
```

Rule 3 ✓

Give your scripts a **descriptive and active name**, e.g.
`06_build_bowtie_index.sh`

- Makes it easier to identify the script of interest

- Helps you to adhere to rule 4

```
01_download_reads.sh
02_run_fastqc.sh
03_trim_reads.sh
04_run_fastqc_trimmed.sh
```

Rule 4 ✓

Make your scripts **specific**, i.e. do not combine many different commands in the same script

- Makes your scripts **modular** (i.e. you can use it for other analyses
- Makes **job submission** more efficient
- Turn your script in a **pipeline** later

Rule 5 ✓

Refer to **directories and variables at the beginning** of the script

- Directories and variables need to be **changed** often
- No need to **search** through the whole script to change them
- **Debugging** is easier

```
#!/usr/bin/env bash
TRIMMED_DIR=~/workdir/trimmed_data
READS_DIR=~/workdir/reads
mkdir -p $TRIMMED_DIR
```

5 simple rules to get started

1. Execute the commands from a script
2. Number scripts based on their order of execution
3. Give your scripts a descriptive and active name
4. Make your scripts specific
5. Directories and variables at the beginning of the script

Further steps

- Version control (git, GitHub, GitLab)
- Pipelines
- Notebooks