



FH WIENER NEUSTADT
BIOTECH CAMPUS TULLN
 – Biotechnology & Digital Future –

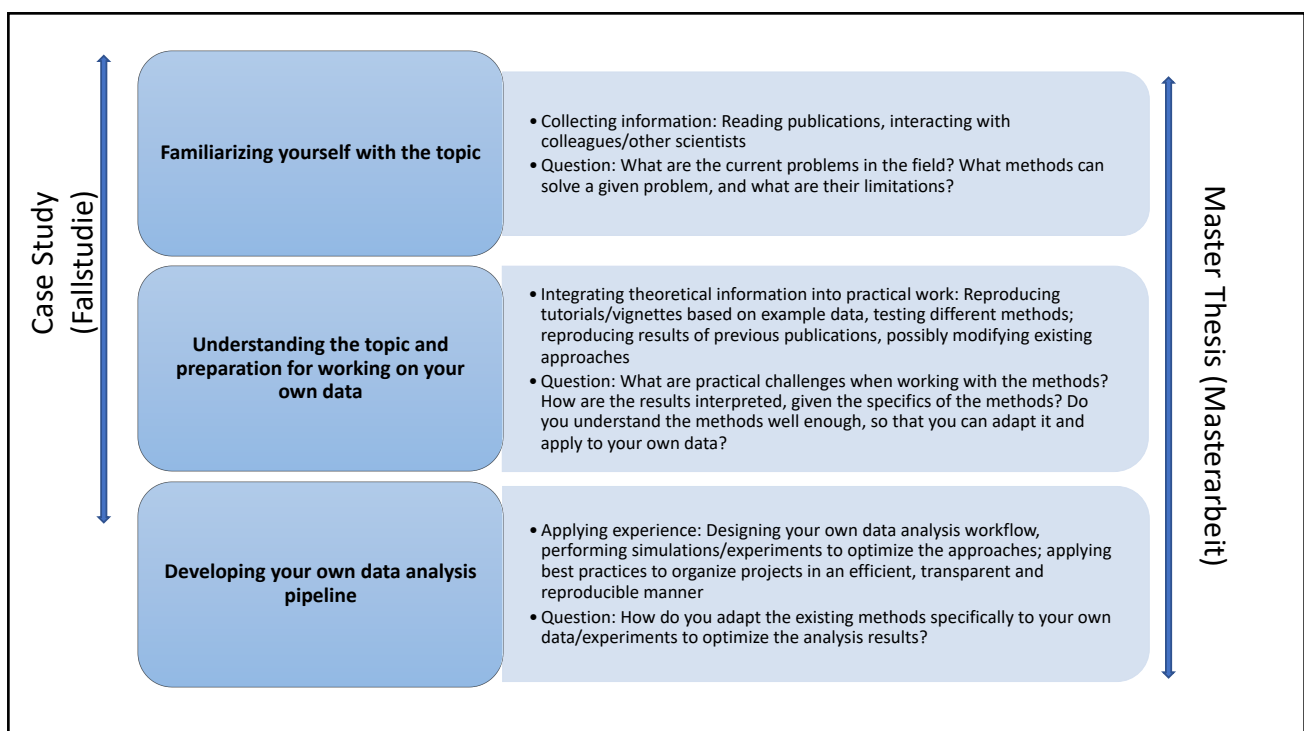
Fallstudie – Case study

16.10.2021

FH Wr. Neustadt, Biotech Campus Tulln

Dmitrij Turaev & Milica Kronic

1



2

Repetition, replication, reproducibility

- **Repetition** is the redoing/repeating your own research
 - repeating measurements, the same/different data sets (assure results not an "accident")
- **Replication** occurs when the research is reproduced/duplicated by a **different** researcher/investigator
 - Using Description of Procedures to reproduce experiments
- "**Reproducibility** refers to the ability of a researcher to duplicate the results of a prior study using the same materials as were used by the original investigator." (U.S. National Science Foundation)
- Repetition and replication ensure **accuracy** of experimental procedures/methods, experimental data and results

3

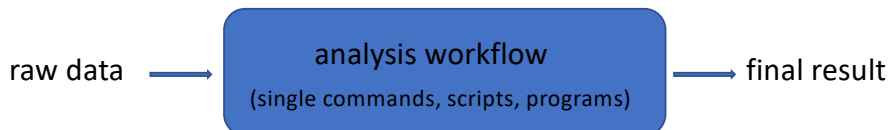
Difficulties in reproducing other studies

- "More than 70% of researchers have tried and failed to reproduce another scientist's experiments, and more than half have failed to reproduce their own experiments." (Baker, 2016)
- Wikipedia: Replication crisis
- scientific papers commonly leave out experimental details essential for reproduction (incomplete Method section, "search details from an author")
- studies showing difficulties with replicating published experimental results (code provided but doesn't work, difficulties with installation, data incomplete, not in the format used in the study)
- "trade-off between the ideals of reproducibility and the need to get the research out while it is still relevant" (Sandve et al. 2013)
- Necessity to establish **routines** that **increase transparency and reproducibility**
- at least be able to reproduce the results yourself
- trust, interest, reuse/citations of your work

4

10+ Rules for [repetition](#)/reproducibility of computational research

1. Keep Track of How Every Result Was Produced



- executable descriptions preferred over manually noting the precise sequence of steps
- As a minimum, you should at least record necessary details on programs, parameters, and manual procedures to reproduce the results

(Sandve et al. 2013)

5

2. Avoid Manual Data Manipulation Steps

- [execution of programs instead of manual procedures to modify data](#)
- Inefficient, error-prone, difficult to reproduce
- Example: If working at the UNIX command line, manual modification of files can usually be replaced by the use of standard UNIX commands or small custom scripts
- If manual operations cannot be avoided, you should as a minimum note down which data files were modified or moved, and for what purpose

6

3. Archive the Exact Versions of All External Programs Used

- to exactly reproduce a given result, it may be necessary to use programs in the exact versions used originally
- a newer version of a program may not even run without modifying its inputs
- store a source code file, or store a full virtual machine image of the operating system and program
- As a minimum, you should note the exact names and versions of the main programs you use

7

4. Version Control All Custom Scripts

- To track evolution of code use a version control system, such as Subversion, Git
- As a minimum, you should archive copies of your scripts from time to time, so that you keep a rough record of the various states the code has taken during development

8

5. Record All Intermediate Results (when possible) in Standardized Formats

- intermediate results **can uncover bugs or faulty interpretations** that are not apparent in the final results
- reveals consequences of alternative programs and parameter choices at individual steps
- **allows parts of the process to be rerun**
- when reproducing results, it allows any experienced inconsistencies to be tracked to the steps where the problems arise
- allows critical examination of the full process behind a result, without the need to have all executables operational
- As a minimum, archive any intermediate result files that are produced when running an analysis

9

6. For Analyses That Include Randomness, Note Underlying Random Seeds

- Many analyses and predictions include some element of randomness, meaning the same program will typically give slightly different results every time it is executed
- given the same initial seed, all random numbers used in an analysis will be equal, thus giving identical results every time it is run
- results to be reproduced exactly the same by **providing the same seed** to the random number generator in future runs
- As a minimum, you should note which analysis steps involve randomness, so that a certain level of discrepancy can be anticipated when reproducing the results

10

7. Always Store Raw Data behind Plots

- If raw data behind figures are stored in a systematic manner, so as to allow raw data for a given figure to be easily retrieved, one [can simply modify the plotting procedure, instead of having to redo the whole analysis](#)
- When plotting is performed using a command-based system like R, it is convenient to also [store the code used to make the plot](#). One can then apply slight modifications to these commands, instead of having to specify the plot from scratch

11

8. Generate Hierarchical Analysis Output, Allowing Layers of Increasing Detail to Be Inspected

- When working with [summarized results](#) (tables, plots), you should as a minimum at least once generate, inspect, and validate the detailed values underlying the summaries.

12

9. Connect Textual Statements to Underlying Results

- As a minimum, you should provide enough details along with your textual interpretations so as to allow the exact underlying results, or at least some related results, to be tracked down in the future

13

10. Provide Public Access to Scripts, Runs, and Results

- all input data, scripts, versions, parameters, and intermediate results should be made publicly and easily accessible
- As a minimum, you should submit the main data and source code as supplementary material, and be prepared to respond to any requests for further data or methodology details by peers

14

11. “Short guide to scientific computational projects”

- LV: Spezielle Werkzeuge für das QM in der Datenanalyse

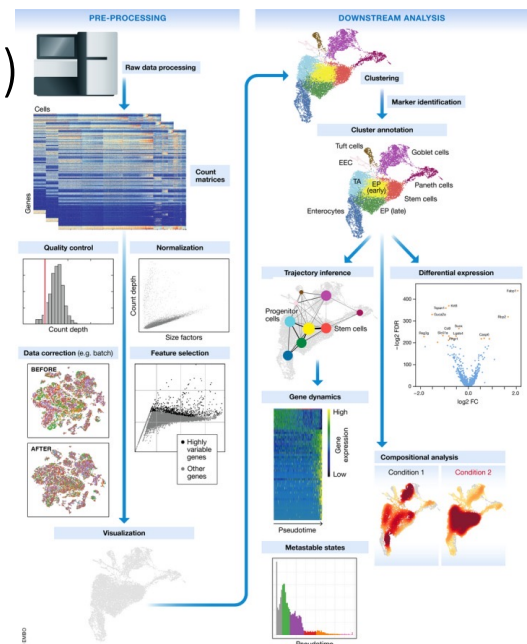
15

Materials for the exam (1/2)

Review > Mol Syst Biol. 2019 Jun 19;15(6):e8746. doi: 10.15252/msb.20188746.

Current best practices in single-cell RNA-seq analysis: a tutorial

Malte D Luecken¹, Fabian J Theis^{2,3}




16

Materials for the exam (2/2)

nature
biotechnology

Integrating single-cell transcriptomic data across
different conditions, technologies, and species

Andrew Butler^{1,2}, Paul Hoffman¹, Peter Smibert¹, Efthymia Papalexi^{1,2} & Rahul Satija^{1,2} 

https://satijalab.org/seurat/articles/pbm3k_tutorial.html

PBMC - Peripheral blood mononuclear cells

17

Tasks and deadlines

- See attached pdf: fallstudie-task-specifications_2021.pdf

18