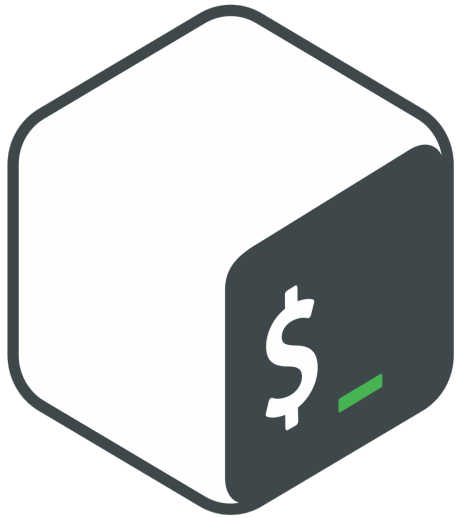
What is a ‘shell’?

The Shell

Unix Shell



BASH

THE BOURNE-AGAIN SHELL

Why bother?

- Automate repetitive tasks
- Manage large sets of information
- Easily combine smaller tasks into larger, more powerful workflows
- Interface with remote systems (including High Performance Computing systems)
- Reproducibility

Nelle's pipeline

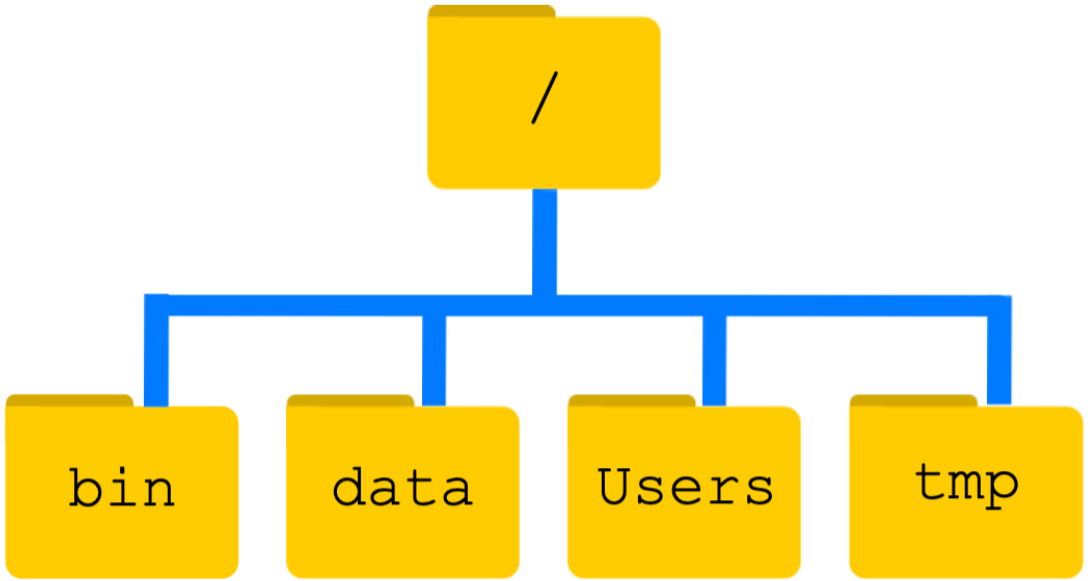
- 1520 samples with 300 data points (protein abundance)
- Need to process this data before analyzing
- ***Highly repetitive task***
- Point-and-click (i.e. GUI-based) software would take hours of hands-on work
- A series of jobs on a high-performance computing cluster (HPCC) could take seconds-minutes

Outline

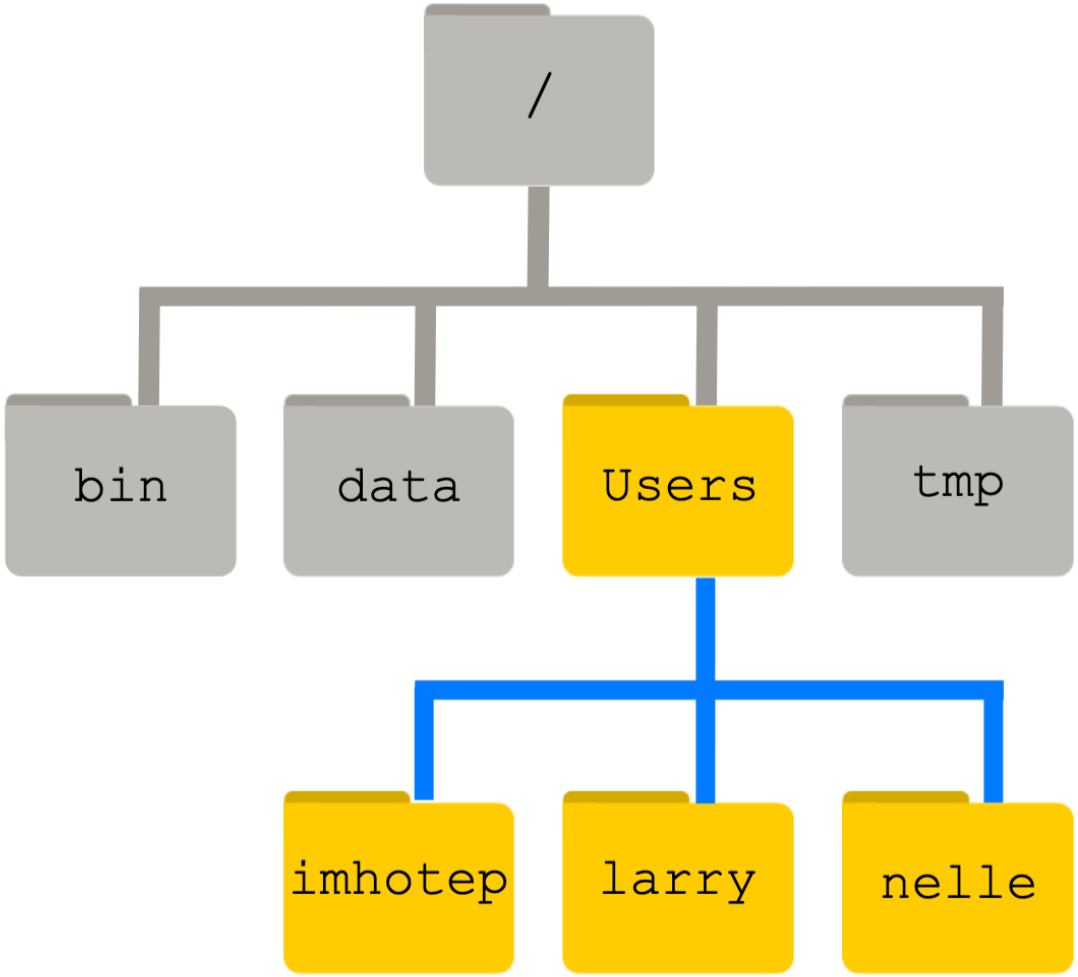
1. Navigating the directory structure
2. Creating files and folders
3. Combining commands to process files
4. Writing and executing scripts

Skipping loops, but the concept will be covered in R modules

Navigating the directory structure

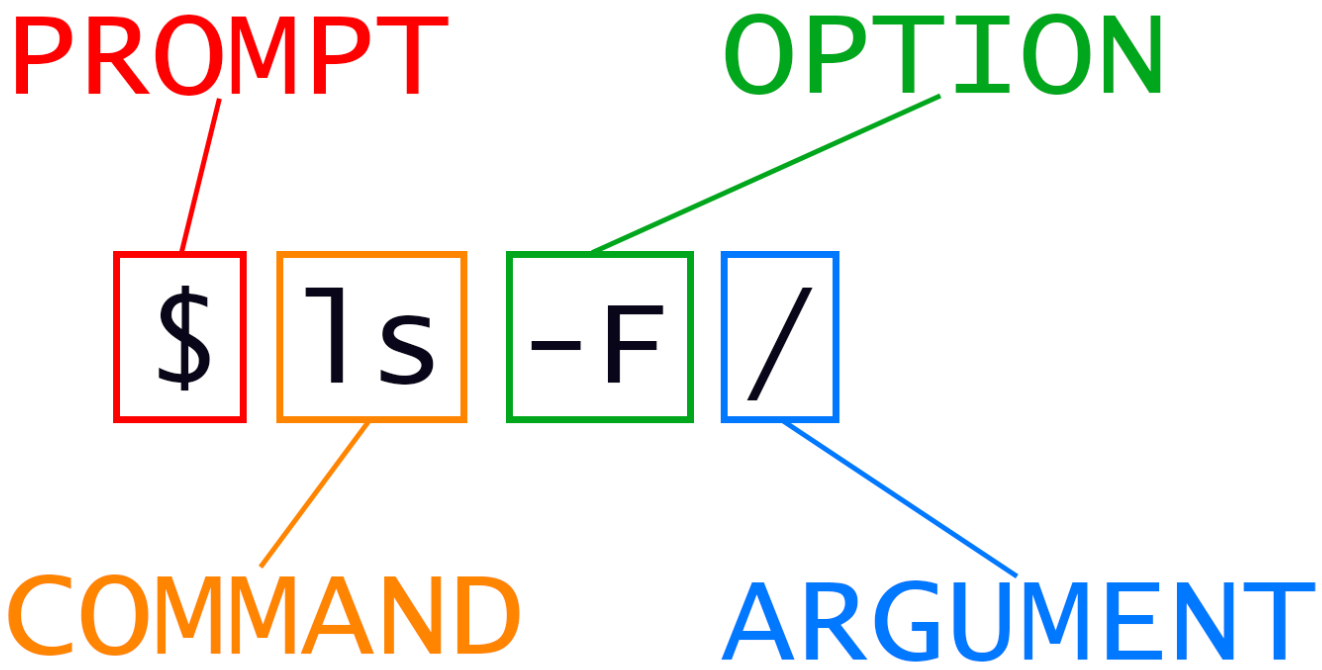


Navigating the directory structure



Navigating the directory structure

Anatomy of a command



Navigating the directory structure

Key commands:

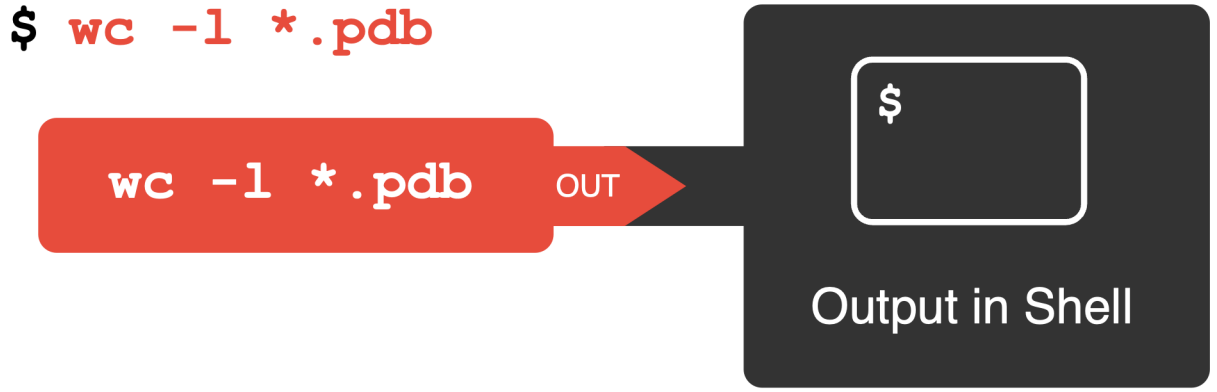
- **`pwd`** – print working directory
- **`cd`** – change directory
- **`ls`** – list contents of directory
- **`man`** – manual entry for command

Creating files and folders

Key commands:

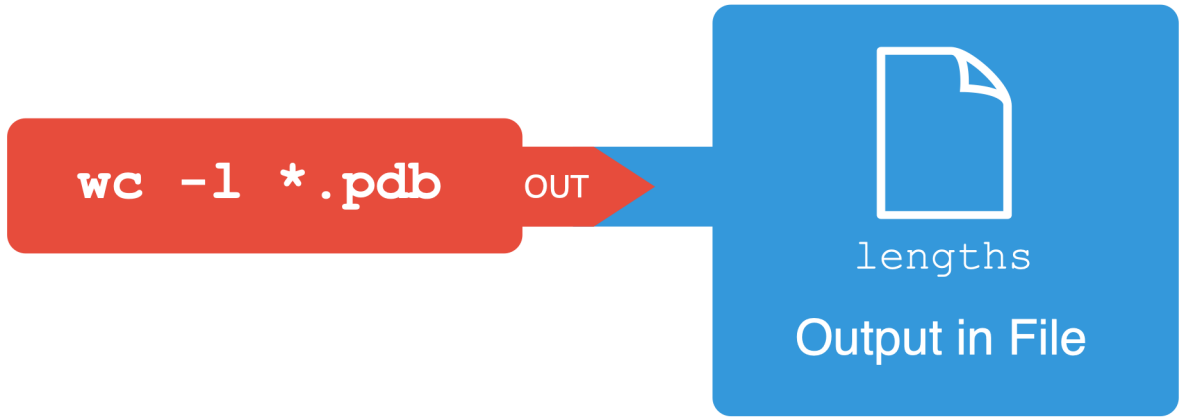
- **`mkdir`** – make directory
- **`nano`** – open nano text editor
- **`less`** – view contents of file
- **`rm`** – remove file or directory
- **`mv`** – move file or directory
- **`cp`** – copy file or directory

Combining commands



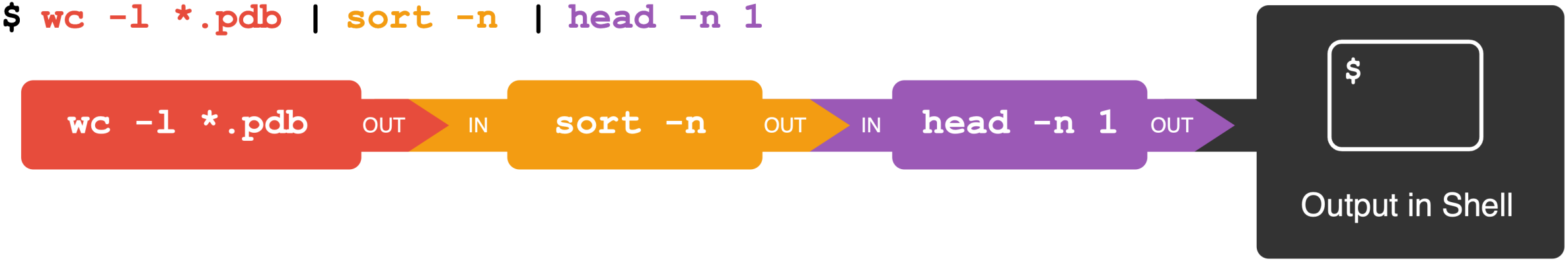
Combining commands

```
$ wc -l *.pdb > lengths
```



Combining commands

```
$ wc -l *.pdb | sort -n | head -n 1
```



Combining commands

Key commands:

- **`wc`** – count words/lines in file
- **`cat`** – print file to standard output/concatenate files
- **`head`** – view first lines of file
- **`tail`** – view last lines of file
- **`sort`** – sort lines of file