Universitätsmedizin Rostock

Microbiome Analysis

31328 Moderne und molekulare Hochdurchsatztechnologien in der medizinischen Grundlagenforschung

Dr. rer. nat. Israel Barrantes

Research Group Translational Bioinformatics (head)
Institute for Biostatistics and Informatics in Medicine and Ageing Research
Rostock University Medical Center
Ernst-Heydemann-Str. 8, 18057 Rostock, Germany

israel.barrantes@uni-rostock.de

October 2025

Overview

- Setup and introduction to the command line
- Background on bioinformatics tools
- Sequence data formats
- Quality control
- Sequence assembly
- Taxonomy assignment

Some Basic Linux Commands

- **ls** - Lists files and directories in the current location; used to see what's in a folder
- **cd** - Changes the current directory; used to navigate between folders (e.g., `cd Documents`)
- **head** - Displays the first few lines of a file; used to preview file contents without opening the entire file
- **wc** - Counts words, lines, and characters in a file; used to get file statistics (e.g., `wc -l` counts lines)
- **grep** - Searches for text patterns within files; used to find specific sequences or keywords (e.g., `grep "ATCG" sequences.fasta`)
- **cat** - Concatenates and displays file contents; used to view entire files or combine multiple files
- **bash** - The command-line shell/interpreter; used to execute commands and run shell scripts

Why Do We Need the Command Line in Bioinformatics?

- **Reproducibility:** Command-line tools and programming enable the creation of reproducible workflows. By writing scripts or workflows, researchers can document and automate their analyses, ensuring that others can replicate their results
- **Pipelines:** Programs talking to each other (pipes)
- **Redirection:** Programs write and read to files
- **Text Streams:** Allow us to both couple programs together and process data without storing huge amounts of data in our computers' memory
- **Modularity:**
  - Modular workflows allow us to experiment with alternate methods and approaches
  - Makes it easier to inspect intermediate results and isolate problematic steps
  - Allows us to choose tools and languages appropriate for specific tasks
  - Modular programs are reusable and applicable to many types of data

Reference: Buffalo, V. (2015) Bioinformatics Data Skills. O'Reilly Media

Reproducibility

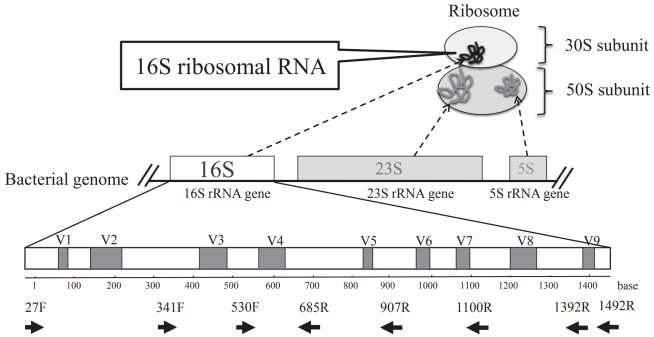
- **Literate programming:** Chunks of programming (analytical code) with human-readable text (comments)
- **Version control:** Tracking changes made to sets of files (of a project), typically program source code, scripts and documentation
- **Environment control:** Versions of all programs (plus libraries, packages, OS) used; archival copies for future reference. Example: sessionInfo()
- **Persistent data sharing:** Collaborative, transparent, accessible science
- **Documentation:** e.g. README file
- Project (data + code + ...) validation
- Command-line tools enable creation of **reproducible workflows** (aka pipelines)
- **Modularity** of the command-line: Possibility of running long pipelines of programs, one after another + piping

Reference: Ziemann et al. Brief Bioinform. 2023 Sep 22;24(6):bbad375

The 16S rRNA Gene for Community Profiling

- **Operational concepts of classification:** OTUs and ASVs
  - **Operational taxonomic units (OTUs):** Consensus sequences from clustering
  - **Amplicon Sequence Variants (ASVs):** Exact sequences
- **Community profiling:** Identifying OTUs and/or ASVs in samples
- 16S rRNA part of the 30S small subunit (SSU) of the prokaryotic ribosome
- All prokaryotes have one (or more) copy of this gene
- Different parts of the gene exhibit different levels of conservation
- Gene $\sim$1540 nt - Not too short to be uninformative; not too long to be unmanageable
- Most 16S rRNA gene is highly conserved between different species
- **Hypervariable regions:** Nine much less conserved (V1–V9)

16S rRNA Gene Structure

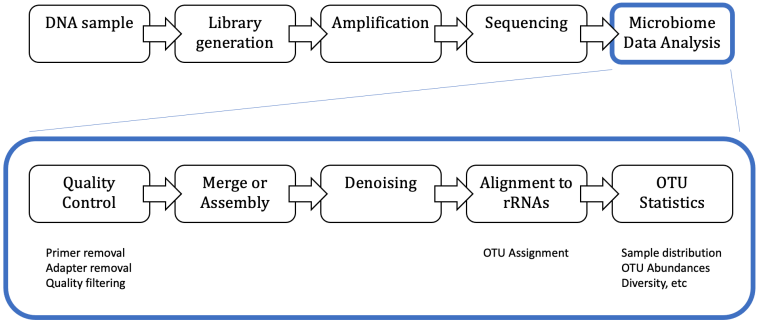


Reference: Fukuda et al. 2016. J UOEH., 38(3):223-32. doi: 10.7888/juoeh.38.223

rRNA Reference Databases

Feature	SILVA	Greengenes	Greengenes2	RDP
Size	∼2M seq.	∼1.2M seq.	∼2.1M seq.	∼3M seq.
Last Update	Rel. 138.1 (2023)	13_8 (May 2013)	2022.10 (Oct 2022)	Rel. 11.5 (2016)
Coverage	Bacteria, archaea, euk.	Bacteria, archaea	Bacteria, archaea	Bacteria, archaea
rRNA genes	SSU + LSU	SSU only (16S)	SSU only (16S)	SSU only
Taxonomy	SILVA tax.	Greengenes tax.	GTDB taxonomy	Bergey's Manual
Best For	General use, euk.	Old studies	General use	Old studies

A Typical Microbiome Analysis Experiment



Sequence Data Formats

FASTA

```
head -10 egfr_flank.fasta
>ENSMUSG00000020122|ENSMUST00000138518
CCCTCCTATCATGCTGTCAGTGTATCTCTAAATAGCACTCTCAACCCCCGTGAACTTGGT
TATTAAAAACATGCCCAAAGTCTGGGAGCCAGGGCTGCAGGGAAATACCACAGCCTCAGT
TCATCAAAACAGTTCATTGCCCAAAATGTTCTCAGCTGCAGCTTTCATGAGGTAACTCCA
GGGCCCACCTGTTCTCTGGT
>ENSMUSG00000020122|ENSMUST00000125984
GAGTCAGGTTGAAGCTGCCCTGAACACTACAGAGAAGAGAGGCCTTGGTGTCCTGTTGTC
TCCAGAACCCCAATATGTCTTGTGAAGGGCACACAACCCCTCAAAGGGGTGTCACTTCTT
CTGATCACTTTTGTTACTGTTTACTAACTGATCCTATGAATCACTGTGTCTTCTCAGAGG
CCGTGAACCACGTCTGCAAT
```

>id, description
sequence….

FASTQ

```
@DJB775P1:248:D0MDGACXX:7:1202:12362:49613
TGCTTACTCTGCGTTGATACCACTGCTTAGATCGGAAGAGCACACGTCTGAA
+
JJJJJIIJJJJJJHIHHHHGHFFFFFFCEEEEEEDBD?DDDDDDBDDDABDDCA
@DJB775P1:248:D0MDGACXX:7:1202:12782:49716
CTCTGCGTTGATACCACTGCTTACTCTGCGTTGATACCACTGCTTAGATCGG
+
IIIIIIIIIIIIIIIHHHHHHFFFFFFEECCCCBCECCCCCCCCCCCCCCCCC
```

@id, description
sequence
+indicates end of sequence
base quality (Phred format)

Hands-on: Working with FASTQ Files

Viewing the first 10 lines of a FASTQ file:

```
! head Platz1_R1.head.fastq
```

Counting total number of lines:

```
! wc -l Platz1_R1.head.fastq
```

Hint: Each read in FASTQ format consists of four lines

Finding specific nucleotide combinations:

```
! grep "AATATT" Platz1_R1.head.fastq | head
```

Counting occurrences:

```
! grep -c "AATATT" Platz1_R1.head.fastq
```

FASTQ Quality

A quality value Q is an integer mapping of p (i.e., the probability that the corresponding base call is incorrect).
$Q_{\text{sanger}} = -10 \log_{10} p$

$$Q_{sanger} = -10 \log_{10} p$$

$$30 = -10 \log_{10} p$$

$$\frac{30}{-10} = \log_{10} p$$

$$p = 10^{-3}$$

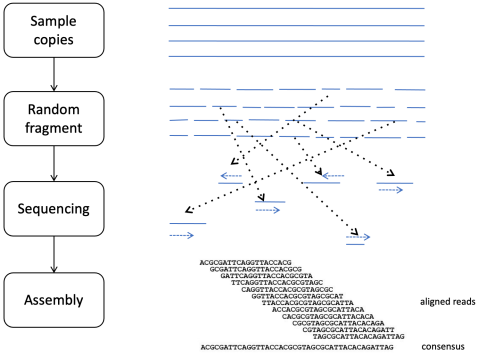
Running FASTQC:

```
! fastqc --quiet Platz1_R1.head.fastq
```

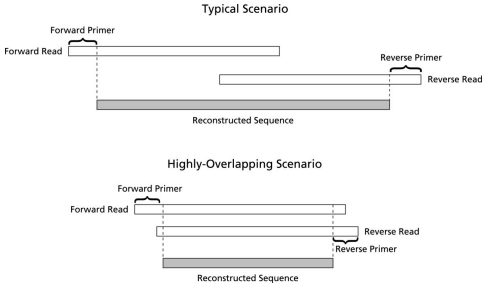
To display the output, download the HTML output from the Files view on the side menu to your local computer (Folder: /course/data2025), and open it in a web browser.

Sequence Assembly

- Reconstructing complete DNA sequences by aligning and merging smaller overlapping DNA fragments
- NGS produces short fragments (e.g. 50-300 bp for Illumina)
- Uses computational algorithms to find overlaps and combine them into longer sequences
- Analog to solving a puzzle with overlapping pieces



Using PANDAseq for Amplicon Assembly



- PANDAseq merges forward and reverse reads by finding their overlapping region
- When reads share identical sequences, it creates a single, longer consensus sequence
- Uses quality scores to decide which base to keep when there are mismatches
- Discards reads that do not overlap sufficiently or have too many errors

Reference: Masella et al. BMC Bioinformatics 13, 31 (2012). doi:10.1186/1471-2105-13-31

Hands-on: Amplicon Assembly with PANDAseq

Running PANDAseq:

```
! pandaseq -f Platz10_R1.head.fastq -r Platz10_R2.head.fastq -w Platz10.fa -g
    log.txt
```

Viewing the output:

```
! head Platz10.fa
```

Counting sequences in FASTA output:

```
! grep -c ">" Platz10.fa
```

Question

What is the rate of FASTA sequences vs FASTQ reads? And what does this tell about our sequencing and assembly quality and efficiency?

Taxonomy Assignment

- **Community profile:** Represents the taxonomic composition of a microbial community obtained from DNA sequencing data. Typically OTU/ASV tables
- **Taxonomy Assignment:** Identifying which organism each DNA sequence belongs to by comparing it against reference databases (SILVA, Greengenes, RDP)
- **OTUs (Operational Taxonomic Units):** Groups of sequences clustered together based on similarity (typically $\geq$97% identity)
- **ASVs (Amplicon Sequence Variants):** Exact unique sequences identified after error correction, providing single-nucleotide resolution. ASVs are now preferred over OTUs for higher precision and reproducibility

Example OTU Table

samples

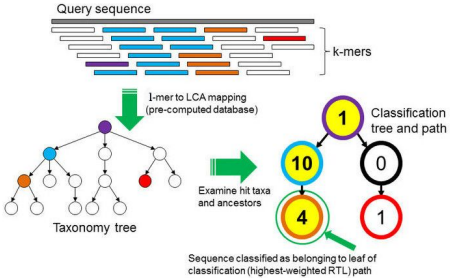
taxa

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
1	# Constructed from biom file																				
2	#OTU ID	Platz18	Platz10	Platz14	Platz2	Platz11	Platz16	Platz15	Platz13	Platz1	Platz3	Platz17	Platz5	Platz7	Platz6	Platz4	Platz12	Platz8	Platz9	taxonomy	
3	18413ece715a299f717980e782e3ed1c	5293	1874	1168	2984	2079	917	2077	815	1843	4594	20519	3310	0	4913	3782	7958	4478	0	d__Bacteria; p__Firmicutes; c__Clostridia; o__Oscillospirales; f__Ruminococcaceae; g__Faecalibacterium	
4	d0a610a019d5ce4e211611d2f9ad7520	3066	867	486	1603	2480	0	1570	523	845	2248	8864	2608	0	2821	2093	3935	1777	0	d__Bacteria; p__Firmicutes; c__Clostridia; o__Oscillospirales; f__Ruminococcaceae; g__Faecalibacterium	
5	437fc004ec290da2418d69d46f821a57	2930	0	0	859	0	0	0	0	0	0	0	0	0	0	0	0	0	0	d__Bacteria; p__Bacteroidota; c__Bacteroidia; o__Bacteroidales; f__Bacteroidaceae; g__Bacteroides	
6	8711aaef804cc58e66f4b21a43d3442a	2592	0	6748	0	0	185	4	0	0	0	0	0	0	8	1961	0	0	0	d__Bacteria; p__Firmicutes; c__Clostridia; o__Oscillospirales; f__Ruminococcaceae	
7	61227bf9cb3b907c5286c7896659b588	2227	434	1208	0	3566	0	0	0	0	0	0	1975	0	173	0	0	0	0	d__Bacteria; p__Bacteroidota; c__Bacteroidia; o__Bacteroidales; f__Bacteroidaceae; g__Bacteroides	
8	536eefdfbba4535eba8401214519b2e2	1602	853	1332	328	0	45	0	0	0	775	0	136	0	0	11151	0	0	0	d__Bacteria; p__Bacteroidota; c__Bacteroidia; o__Bacteroidales; f__Bacteroidaceae; g__Bacteroides	
9	9352c04b5484253922e9338eb71ba883	1529	0	0	5	0	14	4	3	0	0	0	14	10	103	32	254	3	20	d__Bacteria; p__Firmicutes; c__Bacilli; o__Lactobacillales; f__Streptococcaceae; g__Lactococcus	
10	588e76106293f918e78f335a099321f7	1471	0	0	446	0	0	0	0	0	0	0	0	0	0	0	0	0	0	d__Bacteria; p__Bacteroidota; c__Bacteroidia; o__Bacteroidales; f__Bacteroidaceae; g__Bacteroides	
11	ec51863a70cd9a8ab8f28e4375e9da0c	1430	231	0	793	295	0	2645	0	49	0	0	2460	0	328	2	341	9055	1	d__Bacteria; p__Firmicutes; c__Clostridia; o__Oscillospirales; f__Ruminococcaceae; g__Ruminococcus; s__Rum	
12	0a9454a4d21fc3586e7463275ca64e39	1307	0	4036	0	0	79	2	5	0	0	0	0	0	5	1052	0	0	0	d__Bacteria; p__Firmicutes; c__Clostridia; o__Oscillospirales; f__Ruminococcaceae	
13	c2c37a901bbb7868fc23502d83a61c11	1108	257	690	0	4794	0	0	0	0	0	0	1663	0	0	0	0	0	0	d__Bacteria; p__Bacteroidota; c__Bacteroidia; o__Bacteroidales; f__Bacteroidaceae; g__Bacteroides	
14	fbf417e4726cbd91feebbb8818c69573	1014	1477	13014	2041	810	1016	595	1742	6224	1356	6785	2054	4637	5730	20656	2739	10593	5278	d__Bacteria; p__Firmicutes; c__Clostridia; o__Lachnospirales; f__Lachnospiraceae; g__Agathobacter	
15	af2ca0bbcf42a577694f3d91fc29594c	917	0	0	384	0	0	4762	0	0	0	0	0	0	0	1897	843	193	0	d__Bacteria; p__Firmicutes; c__Clostridia; o__Lachnospirales; f__Lachnospiraceae	
16	3520606da0f7747137aee7e2331cc14f	866	787	1996	239	1032	367	561	1245	2719	371	747	955	7424	5337	0	1804	1907	1085	d__Bacteria; p__Firmicutes; c__Clostridia; o__Lachnospirales; f__Lachnospiraceae; g__Blautia	
17	84a0b6767a2fc10ad3fbf446712e5b2d	852	264	439	0	0	7	3	0	2450	4	0	0	30	173	2456	109	0	22	d__Bacteria; p__Firmicutes; c__Bacilli; o__Lactobacillales; f__Streptococcaceae; g__Streptococcus	
18	fa4d87aed60165b86e55598bd3631781	806	349	1033	816	1282	3131	940	2417	5160	1325	4523	1264	0	1806	617	1887	2830	1615	d__Bacteria; p__Firmicutes; c__Clostridia; o__Lachnospirales; f__Lachnospiraceae	
19	4c6c1327d3295eaf9ff519be24059fa9	799	0	735	92	0	0	0	0	0	387	0	191	2	0	6406	0	0	0	d__Bacteria; p__Bacteroidota; c__Bacteroidia; o__Bacteroidales; f__Bacteroidaceae; g__Bacteroides	
20	cec04be4ab4f4cbe3375ae5aa15d2ff7	784	168	2623	285	706	395	496	1853	9551	1777	10190	3131	0	58	12957	5225	9953	19141	d__Bacteria; p__Bacteroidota; c__Bacteroidia; o__Bacteroidales; f__Bacteroidaceae; g__Bacteroides	
21	054bfff28f2ed72af30d4cc85a31701f	775	0	0	3665	0	883	1194	0	788	2085	1360	3054	0	5429	2427	3576	2557	1408	d__Bacteria; p__Firmicutes; c__Clostridia; o__Oscillospirales; f__Ruminococcaceae; g__Subdoligranulum	
22	b497525ee3fcf2183f2667f8648723d9	772	0	3	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0	d__Bacteria; p__Firmicutes; c__Bacilli; o__Lactobacillales; f__Streptococcaceae; g__Lactococcus	
23	88c037a8f27f07c8b4088511eba6679f	755	0	2164	0	1353	0	0	6477	0	0	0	297	0	363	0	0	0	0	d__Bacteria; p__Bacteroidota; c__Bacteroidia; o__Bacteroidales; f__Bacteroidaceae; g__Bacteroides	
24	0521fe8c285174073d2e20a166b972bb	545	4	0	702	0	122	0	11961	4300	5847	0	478	0	2456	0	3606	6131	1385	d__Bacteria; p__Bacteroidota; c__Bacteroidia; o__Bacteroidales; f__Bacteroidaceae; g__Bacteroides	
25	7b194de3e2f4a5746e5cad8adc54bc92	528	0	0	198	0	0	3206	0	0	0	0	0	0	0	0	0	98	0	d__Bacteria; p__Firmicutes; c__Clostridia; o__Lachnospirales; f__Lachnospiraceae	
26	4221b5e418260469d1e30d1bbf803fae	523	771	7846	1106	962	0	427	1108	3313	0	0	1613	0	3194	11635	0	5147	0	d__Bacteria; p__Firmicutes; c__Clostridia; o__Lachnospirales; f__Lachnospiraceae; g__Agathobacter	
27	b58244c9e38034ba80c794b323c36d6d	515	0	0	0	0	0	0	0	0	0	0	0	0	0	346	0	0	0	d__Bacteria; p__Firmicutes; c__Clostridia; o__Oscillospirales; f__Ruminococcaceae; g__Faecalibacterium	
28	c114a0220545f73a2362c1072926a48b	497	188	648	415	1478	1347	563	1399	2660	0	0	958	0	913	0	0	1498	810	d__Bacteria; p__Firmicutes; c__Clostridia; o__Lachnospirales; f__Lachnospiraceae	
29	a5fd013da1d4de3b9261a41b051a07c9	469	0	1247	0	1347	0	374	852	1509	195	0	840	3123	3030	0	922	0	0	d__Bacteria; p__Firmicutes; c__Clostridia; o__Lachnospirales; f__Lachnospiraceae; g__Blautia	
30	8633d67a4c3f1ecef4244fc45f74f767	451	0	0	2158	0	358	854	0	0	1045	0	2476	0	2884	1377	0	0	0	d__Bacteria; p__Firmicutes; c__Clostridia; o__Oscillospirales; f__Ruminococcaceae; g__Subdoligranulum	
31	bcd72c9441a6d1d2f0ef50627f8fc7ff	446	98	1721	201	948	118	373	1363	5446	737	0	2853	0	25	7342	0	5412	10165	d__Bacteria; p__Bacteroidota; c__Bacteroidia; o__Bacteroidales; f__Bacteroidaceae; g__Bacteroides	
32	fb7b2069972a9a26524653070b77dcd0	438	1562	1516	662	485	388	759	4459	2561	2216	4959	711	8985	6811	1156	5581	1450	4641	d__Bacteria; p__Firmicutes; c__Clostridia; o__Lachnospirales; f__Lachnospiraceae; g__Blautia	
33	d51740e49a6fe4984e0afdca13133c23	414	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	d__Bacteria; p__Bacteroidota; c__Bacteroidia; o__Bacteroidales; f__Prevotellaceae	
34	1e1df519b746020db781dbeb323fabb4	406	185	270	0	0	2	2	0	1348	1	0	0	0	126	1311	39	0	49	d__Bacteria; p__Firmicutes; c__Bacilli; o__Lactobacillales; f__Streptococcaceae; g__Streptococcus	
35	834c3607d747f3dc278b590c3ea213c0	403	0	0	285	111	0	395	0	0	0	775	0	0	613	362	0	0	0	d__Bacteria; p__Firmicutes; c__Clostridia; o__Oscillospirales; f__Ruminococcaceae; g__Faecalibacterium	
36	c5646bc185ef619743be4e3b5cf919ff	384	873	1284	153	949	112	843	227	11224	591	0	0	0	2	0	798	7088	0	d__Bacteria; p__Actinobacteriota; c__Actinobacteria; o__Bifidobacteriales; f__Bifidobacteriaceae; g__Bifidobac	
37	d60b3a685294e41bcd11f72d0555181f	344	349	0	860	502	449	153	150	4162	885	0	2266	0	0	0	634	1150	0	d__Bacteria; p__Bacteroidota; c__Bacteroidia; o__Bacteroidales; f__Tannerellaceae; g__Parabacteroides; s__Pa	
38	4c437305a6132032f9c2318be9cabc25	342	0	0	402	0	44	0	8062	2298	3062	0	335	0	1366	0	1885	3105	0	d__Bacteria; p__Bacteroidota; c__Bacteroidia; o__Bacteroidales; f__Bacteroidaceae; g__Bacteroides	
39	357932f2acdb1f12289aff4b2ff932df	333	0	1514	0	1842	0	0	4839	0	0	0	246	0	273	0	0	0	0	d__Bacteria; p__Bacteroidota; c__Bacteroidia; o__Bacteroidales; f__Bacteroidaceae; g__Bacteroides	
40	4b67dcbdc7b4587b187b1fa71a4a942a	301	0	0	76	0	0	0	0	0	0	0	0	0	0	0	0	0	0	d__Bacteria; p__Bacteroidota; c__Bacteroidia; o__Bacteroidales; f__Bacteroidaceae; g__Bacteroides	
41	0b8832af1033a4de11420102cc055535	272	33	0	302	438	318	360	31	947	2508	2971	1546	0	0	0	1736	2861	0	d__Bacteria; p__Bacteroidota; c__Bacteroidia; o__Bacteroidales; f__Rikenellaceae; g__Alistipes	

Taxonomy Assignment with Kraken2

Kraken2 uses the k-mer method:

- Reference database with sequences cut into k-mers (e.g., 30 bases)
- Break new sequence into k-mers
- Check each k-mer in database
- Assign to lowest common ancestor (LCA) if found in multiple organisms
- Decide organism based on where most k-mers point



Reference: Wood & Salzberg, 2014. Genome Biol 15: R46. doi:10.1186/gb-2014-15-3-r46

Running Kraken2 against Greengenes database:

```
! kraken2 --db 16S_Greengenes_k2db --use-names --output output.txt --report
    report.txt --paired Platz10_R1.head.fastq Platz10_R2.head.fastq
```

Inspecting results:

```
! cat report.txt
```

Question

What are the most predominant genera in your personal Illumina runs?

Kraken2 Report Output Format

The output report from Kraken2 consists of the following fields:

- Percentage of fragments covered by the clade rooted at this taxon
- Number of fragments covered by the clade rooted at this taxon
- Number of fragments assigned directly to this taxon
- A rank code: (U)nclassified, (R)oot, (D)omain, (K)ingdom, (P)hylum, (C)lass, (O)rder, (F)amily, (G)enus, or (S)pecies
- NCBI taxonomic ID number
- Indented scientific name

Generating the OTU Table

Execute Kraken2 with all Illumina sequencing pairs:

```
! bash getkrakenreports.sh
```

Merge all reports with kraken-biom:

```
! kraken-biom *.report --fmt tsv -o mbtmicrobiome20251022.tsv
```

View the OTU table:

```
! cat mbtmicrobiome20251022.tsv
```

OTU taxonomic information can be retrieved from the NCBI Taxonomy database

Aims

- How do the samples group with each other? Principal Components
- How much diversity is there in a sample? Alpha diversity
- How does the diversity differ between samples? Beta diversity
- What is the taxonomic content of the sample?
- Rarefaction: Adjusting for differences in library sizes across samples to aid comparisons

Reference: Willis A. 2019. Front. Microbiol. 10: 2407, doi:10.3389/fmicb.2019.02407

Why R for Bioinformatics?

- Freely available
- User-friendly resources to learn R
- R editors and integrated development environments (IDEs)
- Active user community
- Large code repositories: CRAN, Bioconductor, github, gitlab
- Ready-made packages and functions for Bioinformatics
- Reproducibility: Code, library versions, Rmarkdown, Jupyter notebooks

Install Bioconductor and phyloseq:

```
install.packages("BiocManager")
BiocManager::install("phyloseq")
```

Load required libraries:

```
# load ggplot2 library (graphics)
library(ggplot2)

# loading phyloseq library (microbiome analysis)
library(phyloseq)
```

Loading Input Files

```
# OTU data
InputBiomFile <- "mbtmicrobiome2024.biom"

# Samples' metadata
InputMapFile <- "sample-metadata-2024.tsv"

# prepare phyloseq object by loading both files
BiomData <- import_biom(InputBiomFile, parseFunction = parse_taxonomy_
    greengenes)
SampleData <- import_qiime_sample_data(InputMapFile)
```

View metadata:

```
head(SampleData)
```

Creating a Phyloseq Object

Merge OTU and sample data:

```
# create phyloseq object by merging OTU and sample data
ExperimentPhyloseqObject <- merge_phyloseq(BiomData, SampleData)
```

Check object properties:

```
# checking the features of our original microbiome data
ExperimentPhyloseqObject
```

Subsetting and Filtering Data

```
# create a temporary phyloseq object for working
psTemp <- ExperimentPhyloseqObject

# Prune OTUs with low abundances from all samples
psTemp <- prune_taxa(taxa_sums(psTemp) > 100, psTemp)

# Prune samples with no metadata
psTemp <- subset_samples(psTemp, Gender != "U")

# checking the features of our microbiome data
psTemp
```

Question

What are the differences between the ExperimentPhyloseqObject and psTemp objects?

Sample Ordination

- Ordination is a visualization technique that places samples in 2D or 3D space based on their microbial community similarity
- Samples with similar microbiomes appear close together; different ones are far apart
- Ordination shows quickly if samples cluster by treatment, disease state, or other factors—revealing biological patterns in your data at a glance
- **PCoA (Principal Coordinates Analysis):** Works with ecological distances like Bray-Curtis dissimilarity, which better captures compositional differences between communities

References: Bray & Curtis (1957). Ecol Monogr, doi:10.2307/1942268
Ramette (2007). FEMS Microbiol Ecol, doi:10.1111/j.1574-6941.2007.00375.x

Hands-on: Sample Ordination

```
# Calculate distance and ordination
iDist <- distance(psTemp, method="bray")
iMDS  <- ordinate(psTemp, distance=iDist)

# plot sample ordination, e.g. by Gender
plot_ordination(psTemp, iMDS, color="Gender")
```

Question

Are there any clear separations between the gender groups? Why/Why not?

Ordination by other metadata (e.g. "Pet"):

```
plot_ordination(psTemp, iMDS, color="Pet") +
  geom_text(aes(label=SampleID), vjust = -1)
```

Phylum and Genus Barplots

- Microbial communities are usually represented by bar plots showing the relative abundance (percentage) of different bacterial groups
- Each color represents a different taxon, and bar height shows how much of the community it comprises
- After taxonomy assignment, sequences are grouped by taxonomic level and counted. Counts are converted to percentages so all bars add up to 100%
- Provides an intuitive visual summary of community composition
- Typically only the top 10-20 most abundant taxa are shown individually; rare taxa are grouped as "Other"

References: Caporaso et al. (2010). Nature Methods, doi:10.1038/nmeth.f.303
McMurdie & Holmes (2013). PLoS One, doi:10.1371/journal.pone.0061217

Rarefaction

- Normalization technique that randomly subsamples all samples to the same sequencing depth (number of reads) to enable fair comparison
- Different samples often have different total read counts due to technical variation in sequencing
- All samples are randomly downsampled to match the sample with the fewest reads
- **However:** Some argue rarefaction discards valuable data and recommend alternative normalization methods (e.g., CSS, DESeq2), particularly for differential abundance testing

References: Hughes et al. (2001). Appl Environ Microbiol, doi:10.1128/AEM.67.10.4399-4406.2001
McMurdie & Holmes (2014). PLoS Comput Biol, doi:10.1371/journal.pcbi.1003531

Absolute abundances at the Phylum and Genus levels:

```
### Phylum
plot_bar(psTemp, "SampleID", fill="Phylum")

### Genus
plot_bar(psTemp, "SampleID", fill="Genus")
```

Note

These plots show that the samples are not directly comparable. Hence rarefaction is needed to adjust for differences in library sizes.

Hands-on: Abundances After Rarefaction

```
# initializes the random number generator to ensure reproducibility
set.seed(1212)

# Rarefaction to an even depth
ps.rarefied <- rarefy_even_depth(psTemp)

# Agglomerate the data to a single taxonomic level, e.g. Phylum
ps.rarefied.glom <- tax_glom(ps.rarefied, "Phylum")

# Plot abundances
plot_bar(ps.rarefied.glom, "SampleID", fill="Phylum")
```

Question

What are the differences between the abundance plots before and after rarefaction?

Using Metadata

```
### Abundances per category, e.g. "Gender"
# Merge samples by the selected category
mergedGP <- merge_samples(psTemp, "Gender")

# Rarefaction to an even depth
ps.rarefied <- rarefy_even_depth(mergedGP)

# Agglomerate to Phylum level
ps.rarefied.glom <- tax_glom(ps.rarefied, "Phylum")

# Plot abundances for the example category "Gender"
plot_bar(ps.rarefied.glom, fill="Phylum")
```

Diversity

- **Diversity:** How many different species are present and how evenly distributed they are in microbial communities
- **Alpha diversity:** Diversity **within** a single sample
  - Common metrics: species richness, Shannon index, observed ASVs/OTUs
  - Shannon index combines richness and evenness: higher values = more diverse communities
- **Beta diversity:** Diversity **between** samples
  - Measured using distance metrics like Bray-Curtis dissimilarity or UniFrac distances
  - Used to create ordination plots (PCoA, PCA)
- Low alpha diversity may indicate dysbiosis; high beta diversity between groups suggests different conditions are shaping distinct communities

References: Magurran (2004). Measuring Biological Diversity, doi:10.1002/9780470999738
Lozupone & Knight (2005). Appl Environ Microbiol, doi:10.1128/AEM.71.12.8228-8235.2005

Hands-on: Alpha Diversity by Sample

Shannon diversity on individual samples:

```
plot_richness(psTemp, x = "SampleID", measures = c("Shannon"))
```

Shannon diversity by Gender:

```
plot_richness(psTemp, x = "Gender", color = "Gender", measures = c("Shannon")
    )
```

Enhanced Diversity Visualization

```
# Assign our plot to a variable
Our_Richness_plot <- plot_richness(psTemp, x = "Gender",
                                   color = "Gender",
                                   measures = c("Shannon"))

# Improving our plot by adding features
Our_Richness_plot +
  geom_boxplot(data = Our_Richness_plot$data,
               aes(x = Gender, y = value, color = Gender),
               alpha = 0.1) + # boxplot
  labs(title = "Richness of the gut microbiome (Shannon alpha diversity)",
       subtitle = "MBT Class 2024") + # title and subtitle
  theme(axis.text.x = element_text(angle = 0, hjust = 0.5)) # x-axis labels
```

Quiz: Steps for the Analysis of Microbiome Data

Statement	**T**	**F**
Alignment to rRNA sequences from standard databases	T	F
microRNA identification	T	F
RNA structure prediction	T	F
Quality control	T	F
Denoising or quality filtering	T	F

Quiz: Assembly in Microbiome Analysis Means?

Statement	**T**	**F**
Discard sequences with poor FASTQ quality	T	F
Comparison (alignment) of the amplicon sequences versus known rRNAs from ribosomal rRNA databases	T	F
Preparing OTU tables by counting the number of rRNA matches present for each taxa	T	F
Merging fragments (in FASTQ format) into whole sequences (in FASTA format)	T	F
Building a small database of ribosomal RNAs	T	F

Quiz: Abundance (Composition) Plots Show:

Statement	**T**	**F**
The phyla and genera compositions simultaneously for all samples in the same plot	T	F
The genera (genus) and species composition simultaneously of a microbial community	T	F
The abundance of all taxonomic levels at the same time for every individual sample	T	F
The abundance of all taxonomic levels at the same time for every sample group	T	F
The composition of a given single taxonomic level for a microbial community	T	F

Quiz: R for Bioinformatics

Statement	**T**	**F**
Requires a commercial licence for use	T	F
Large code repositories	T	F
Jupyter Notebooks are limited to Python and hence cannot be used with the R programming language	T	F
Ready-made packages and functions for bioinformatics	T	F
Reproducibility depends on R library versions	T	F

Quiz: Microbial Communities

Statement	**T**	**F**
Rarefaction is a method that adjusts for differences in library sizes across samples to aid comparisons	T	F
The principal components method is used to visualize how samples group together	T	F
Genera composition tells the abundance of species in a sample or group of samples	T	F
The taxonomic content of a sample (or group) can only be studied using known taxa	T	F
Alpha diversity measures can be seen as a summary statistic of a single population (within-sample diversity)	T	F