

A Practical Guide to Holo-Omics

The Center for Evolutionary Hologenomics

Contents

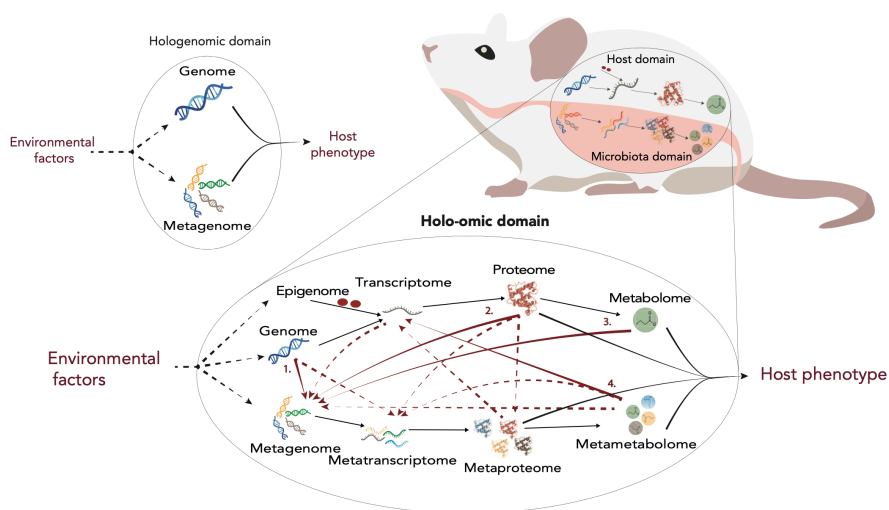
About this guidebook	7
I INTRODUCTION	11
1 Introduction to holo-omics	13
1.1 Omic layers	14
2 Study design considerations	17
2.1 Hologenomic complexity	18
2.2 Control of variables	20
2.3 Molecular resolution	21
2.4 Spatiotemporal factors	24
2.5 Explanatory and response variables	25
II LABORATORY PROCEDURES	29
3 About labwork	31
4 DNA/RNA extraction	33
5 Protein/metabolite extraction	37
6 Sequencing library preparation	39
6.1 Host genomics and microbial metagenomics	41
6.2 Host transcriptomics	42
6.3 Microbial metatranscriptomics	45
III BIOINFORMATIC PROCEDURES	47
7 About bioinformatics	49
7.1 Prepare your shell environment	49

7.2	Example data for bioinformatics	51
7.3	Using snakemake for workflow management	51
8	Sequencing data preprocessing	53
9	Host genomics (HG) data processing	55
9.1	Host reference genome	55
9.2	Host genome resequencing	55
10	Microbial metagenomics (MG) data processing	57
10.1	Reference-based	58
10.2	Assembly-based	58
10.3	Genome-resolved	62
11	Host transcriptomics (HT) data processing	67
11.1	Reference-based host transcriptomics (HT) data processing . . .	67
11.2	Reference-free host transcriptomics (HT) data processing	69
12	Microbial metatranscriptomics (HT) data processing	71
12.1	Reference-based microbial metatranscriptomics (MT) data processing	71
12.2	Reference-free microbial metatranscriptomics (MT) data processing	73
13	Host proteomics (HP) data processing	75
14	Microbial metaproteomics (MP) data processing	77
IV	STATISTICAL PROCEDURES	79
15	About statistics	81
15.1	Prepare your R environment	82
15.2	Example data for statistics	83
16	Single omic analyses	85
17	Data transformations	87
17.1	Transformations to normalise data	87
17.2	Transformations to account for missing observations	87
17.3	Transformations to account for compositional nature of the data	88
17.4	Transformations to account for scaling	88
18	Unsupervised exploration	89
18.1	Cluster analysis	89
18.2	Dimension reduction and ordination	90
19	Supervised analysis	93
19.1	Regression methods	93

<i>CONTENTS</i>	5
19.2 Classification methods	94
20 Multi-omic integration	95
21 Multi-staged integration	97
22 Meta-dimensional integration	99
 V RESOURCES	 101
23 Useful links	103
24 References	105

About this guidebook

The practical guide to holo-omics is under construction, and its contents are still preliminary. We expect to have an initial complete version by May 2023.



Holo-omics overview. Modified from Nyholm et al. 2020 [Nyholm et al., 2020]

The **practical guide to holo-omics** is a compilation of methodological procedures to generate, analyse and integrate holo-omic data, i.e., multi-omic data jointly generated from hosts and associated microbial communities [Nyholm et al., 2020, Limborg et al., 2018]. This guide extends the contents of the article “**A practical introduction to holo-omics**”, which aims at guiding researchers to the main critical steps and decision points to perform holo-omic studies. While the article focuses on discussing pros and cons of using multiple available options, the aim of this guide is to compile protocols and pipelines to be implemented by researchers. The **practical guide to holo-omics** is presented in two formats:

- Website (<http://www.holo-omics.science/>)
- PDF document (http://www.holo-omics.science/holo_omics_workbook.pdf)

This guide is presented as a final output of the H2020 project HoloFood. More information about this EU Innovation Action that ran between 2019 and 2023 can be found in the HoloFood Website and the CORDIS website.

Contents

- **Introduction:** general information about holo-omics, employed data types and study design considerations.
- **Laboratory procedures:** methods and procedures for generating raw omic data of hosts and microbial communities.
- **Bioinformatic procedures:** methods and procedures for processing raw omic data into quantitative datasets to be analysed through statistics.
- **Statistical procedures:** methods and procedures for analysing and integrating holo-omic data.

Protocols, exercises and tutorials

This guide contains example data and bits of code (mostly shell and R) to reproduce data generation and analysis procedures. Code boxes look like the following:

```
shao4d_perm <- shao4d %>%
  tax_transform("identity", rank = "genus") %>%
  dist_calc("aitchison") %>%
  dist_permanova(
    variables = c("birth_mode", "sex", "number_reads"),
    n_perms = 99, # you should use more permutations in your real analyses!
    n_processes = 1
  )
#> Dropping samples with missings: 15
#> 2022-11-24 01:15:20 - Starting PERMANOVA with 99 perms with 1 processes
#> 2022-11-24 01:15:21 - Finished PERMANOVA

shao4d_perm %>% perm_get()
#> Permutation test for adonis under reduced model
#> Marginal effects of terms
#> Permutation: free
#> Number of permutations: 99
#>
#> vegan::adonis2(formula = formula, data = metadata, permutations = n_perms, by = by,
#>
#>           Df SumOfSqs      R2      F Pr(>F)
#> birth_mode    1    10462 0.09055 29.3778  0.01 **
#> sex           1     402 0.00348  1.1296  0.31
```



```
#> number_reads 1 1117 0.00967 3.1364 0.01 **
#> Residual 287 102209 0.88462
#> Total 290 115540 1.00000
#> ---
#> Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Authors

Antton Alberdi

Antton is an Associate Professor at the University of Copenhagen whose research is focused on understanding how animal-microbiota interactions shape basic and applied biological processes. Antton is the corresponding author of the **Practical Guide to Holo-omics**.

antton.alberdi@sund.ku.dk | www.alberdilab.dk

Morten Limborg

Text to be included.

Raphael Eisenhofer

Text to be included.

- Morten T Limborg, University of Copenhagen
- Iñaki Odriozola, University of Copenhagen
- Jacob A Rasmussen, University of Copenhagen

Protocol and script contributors

- Carlotta Pietroni, University of Copenhagen
- Jorge Langa, University of Copenhagen

Other relevant people

- Tom Gilbert (HoloFood project coordinator), University of Copenhagen
- Anna Fotaki (HoloFood project manager), University of Copenhagen

How to cite this work

Instructions to cite this work will be eventually added.

Acknowledgement



This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 817729.

Part I

INTRODUCTION

Chapter 1

Introduction to holo-omics

Why do we need holo-omics?

Every multicellular organism is a host ‘environment’ for which microbes pass through, persist, replicate, and/or influence the host phenotype. Evidence, collected from rainforest swamps to research labs, and from farm stables to patient bedsides, has made it clear that no fauna or flora live alone. Although each have their own peculiar characteristics, animals and plants are incontrovertible assemblages of multiple lifeforms. They compositionally form holobionts with their diverse microbial associates, whether they are transient or stably present [Theis et al., 2016]. Holobionts can thus change in time and space, and the collective gene catalog of a holobiont in turn forms a hologenome, which can yield variation in phenotypes with fitness, performance, or disease consequences. The prefix “holo” derives from the Greek word holos for entire or whole. Holobiont and hologenome are thus structural terms that help us view and study biological systems in an integrated community context, that are subject to diverse ecological and evolutionary forces with harmful, helpful, or harmless consequences [Rosenberg and Zilber-Rosenberg, 2013]. The terms also recognize that hosts often outsource or intertwine metabolism to stable or transient microbial associates, and that hosts have evolved a gradient of dependencies and antagonisms with microorganisms in or on their surfaces and surroundings across the plant and animalia kingdoms.

What is holo-omics?

Holo-omics refers to the methodological approach that jointly generates and analyses multi-omic data from hosts and associated microbial communities [Nyholm et al., 2020]. Holo-omics leverage current knowledge and methods in the fields of molecular biology and microbiology into a novel framework integrating molecular data including genomes, transcriptomes, epigenomes, proteomes, and metabolomes for analyzing host organisms and their gut microbiota as inter-

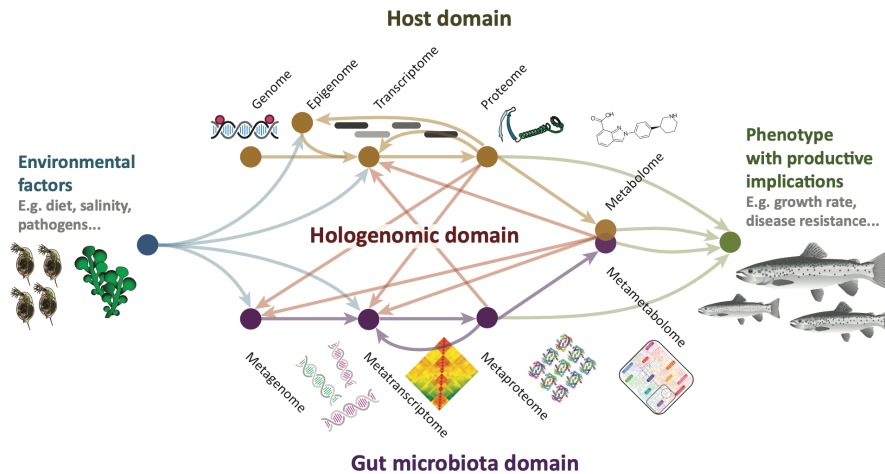
connected and coregulated systems. The advantage of holo-omics is that it is supposed to overcome the limited functional insights of current analytical strategies by simultaneously considering the holobiont at multiple molecular levels. This involves deciphering interactions between not only the host genome but also its epigenome and transcriptome, as well as its microbial metagenome and metatranscriptome. Studies would ideally also incorporate analyses of the associated proteomes and metabolomes, and metaproteomes and metametabolomes, to fully recover the functional pathways controlling the observable phenotype of a host organism. Successful integration of such data into a holo-omic framework will reveal mechanisms such as how host genomes regulate the composition of the microbial community, or, conversely, how specific microbes interact to control host gene expression patterns. Finally, the holo-omic approach to study host-microbiota interactions relies on three major assumptions of the study system:

1. Host-associated microorganisms interact not only with each other but also with their host [Fischer et al., 2017].
2. These interactions affect, either positively or negatively, central biological processes of hosts and microorganisms [Wu and Wu, 2012].
3. The interplay can be traced using biomolecular tools.

Contents of this section were created by Antton Alberdi and Morten Limborg.

1.1 Omic layers

Nucleic acid sequencing and mass spectrometry technologies that enable tracking the biomolecular pathways linking host and microbial genomic sequences with biomolecular phenotypes by generating (meta)transcriptomes, (meta) proteomes, and (meta)metabolomes. The same technologies also enable epigenomic and exposomic profiling, which can further contribute to disentangling the biochemical associations between host-microbiota-environment interactions and their effect on host phenotypes.



Overview of omic layers. Modified from Limborg et al. 2018 [Limborg et al., 2018].

In this workbook we consider seven omic layers that require specific data generation and analysis strategies before integrating them into multi-omic statistical models:

- Nucleic acid sequencing-based
 - Host genomics - **HG**
 - Host transcriptomics - **HT**
 - Microbial metagenomics - **MG**
 - Microbial metatranscriptomics - **MT**
- Mass spectrometry-based
 - Host proteomics - **HP**
 - Microbial metaproteomics - **MP**
 - (Meta)metabolomics - **ME**

Acknowledging the distinct biological and structural characteristics of these seven omic layers is essential to design experiments and analytical pipelines for better solving the complex puzzle of host-microbiota interactions.

Host genomics (HG)

Contents to be added

Host transcriptomics (HT)

Contents to be added

Microbial metagenomics (MG)

Contents to be added

Microbial metatranscriptomics (MT)

Contents to be added

Host proteomics (HP)

Contents to be added

Microbial metaproteomics (MP)

Contents to be added

(Meta)metabolomics (ME)

Contents to be added

Chapter 2

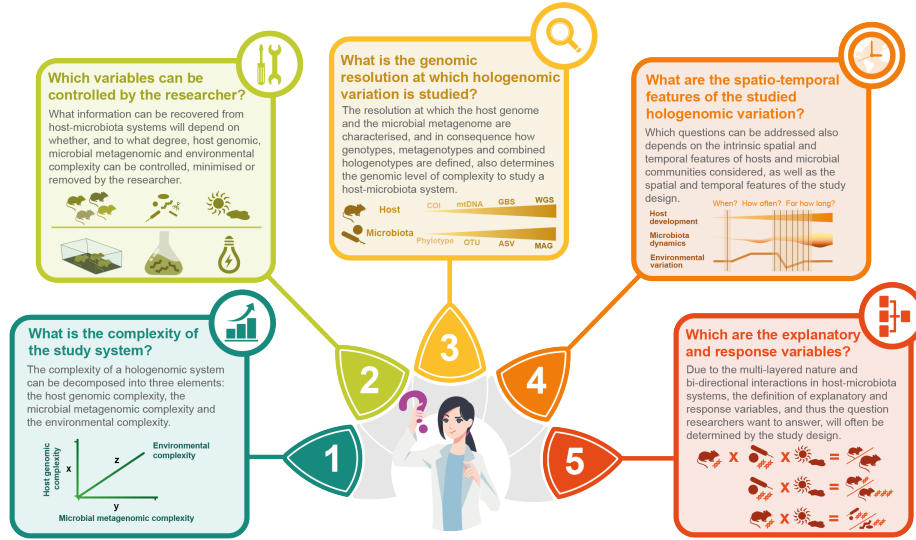
Study design considerations

The contents of this section have been extracted and modified from the article Disentangling host–microbiota complexity through hologenomics published in *Nature Reviews Genetics* in 2022 by the authors of the **Holo-omics Workbook**.

Holo-omic approaches can be used to understand how the combined features of hosts and microorganisms shape biological processes relevant for hosts (such as adaptation), for microorganisms (such as meta-community dynamics) or both [Alberdi et al., 2022].

Depending on the aims and features of the study system, holo-omics can be implemented using different study designs, model systems and techniques. This landscape of possibilities is shaped around five essential questions that need to be considered when designing and interpreting hologenomic studies, which relate to five core topics:

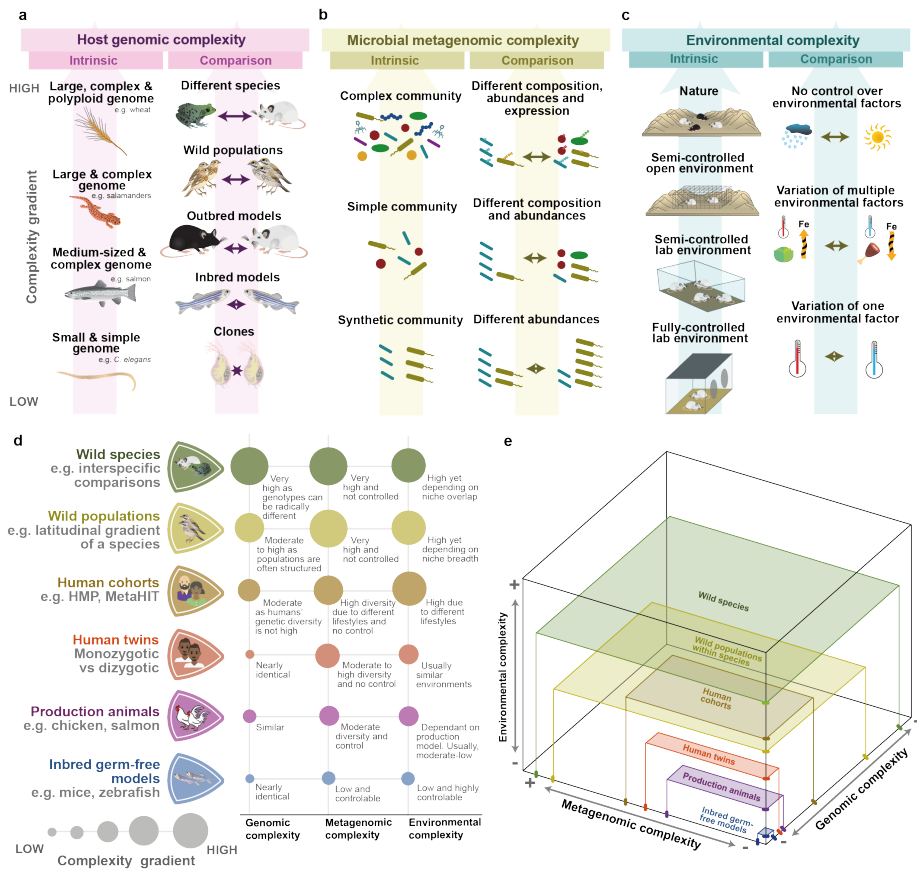
1. **Hologenomic complexity**
2. **Control of variables**
3. **Molecular resolution**
4. **Spatiotemporal factors**
5. **Explanatory and response variables**



2.1 Hologenomic complexity

The contents of this section have been extracted and modified from the article Disentangling host-microbiota complexity through hologenomics published in *Nature Reviews Genetics* in 2022 by the authors of the **Holo-omics Workbook**.

Hologenomic complexity can be broadly defined as the amount of information relevant to the study that the biological system under analysis contains and it can be decomposed into three major elements: host genomic, microbial metagenomic and environmental complexity [Alberdi et al., 2022]. Within each of these elements, two sources of complexity can be defined: the intrinsic complexity of the system under study, including host genome size and number of bacterial genomes, and the complexity introduced by the degree of difference between the organisms under comparison such as gene expression differences versus distinct genomes.



Decomposition of hologenomic complexity. (a-c) The design and interpretation of hologenomic studies depend on the host genomic (part a), microbial metagenomic (part b) and environmental (part c) complexity of the system under study. Within each axis of complexity, two types of gradients can be defined based on whether the features are intrinsic to the system or introduced by the researcher through the selection of groups under comparison. (d) Six examples of study systems with different levels of genomic, metagenomic and environmental complexity. (e) Three-dimensional representation of the complexity of the examples. The area of the plain represents the combined host genomic and microbial metagenomic complexity of the system, while the height represents the environmental complexity. The combined three-dimensional volume represents the overall hologenomic complexity of the system. HMP: Human Microbiome Project.

2.2 Control of variables

The contents of this section have been extracted and modified from the article Disentangling host–microbiota complexity through hologenomics published in *Nature Reviews Genetics* in 2022 by the authors of the **Holo-omics Workbook**.

Controlling the complexity of hologenomic variables is essential for addressing specific research questions. Broadly speaking, the more detailed and mechanistic the question under study, the greater the required control. For instance, research on specific biomolecular processes using laboratory models will require a higher level of control than studying biogeographical patterns of host–microbiota interactions in wild organisms. The control of hologenomic variables can be achieved through a number of strategies.

2.2.1 Controlling host genomes

The control over host genomic complexity largely depends on the model organisms studied and the technical approaches employed. In laboratory organisms that can reproduce asexually, such as water fleas (*Daphnia*, Crustacea) and Lamiaceae plants, absolute control over host genotypes can be achieved by using clonal organisms [Mushegian et al., 2019]. When clones cannot be used, inbred laboratory animals can provide a high level of genomic homogeneity. The use of groups of genetically homogeneous hosts allows the effects of contrasting environmental conditions or specific microbial communities to be compared. Clonal and inbred models also enable the effects of a specific host genetic factor to be studied in a controlled genomic background through the application of targeted techniques for modulating gene expression (such as RNA-mediated interference) or for genomic engineering (such as CRISPR–Cas9). Working with humans and wild organisms does not enable such a degree of control over the genotypes studied unless in vitro models, such as organ-on-a-chip co-cultures of animal tissues and microbial communities, are generated⁶². When this level of control is not possible, coarse control over host genotypes can be achieved through contrasting animals from different populations or from closely related species⁶³, while greater control can be achieved through comparing individuals across different degrees of kinship, such as monozygotic versus dizygotic twins³⁸, and family members to other individuals⁶⁴.

2.2.2 Controlling microbial metagenomes

Control over microbial metagenomic complexity is usually achieved through modulating microbial communities. Some strategies, such as modification of dietary regimes or the administration of microbiota-targeted additives or prebiotics, aim to modify microbial ecosystems by changing nutrient availability. However, unless compounds that match unique enzymatic capabilities of specific microorganisms are used, it is difficult to accurately modulate the microbiota

owing to the complexity of ecological relationships among microorganisms. Alternative approaches to modify microbial communities include inoculation of target bacteria (such as probiotics) and faecal microbiota transplantation. The efficacy and accuracy of these methods is also variable; there is no guarantee that inoculated bacteria will establish or modulate the microbiota, while transplantation does not enable accurate control over the microbial community introduced or the secondary elements that are transplanted along with bacteria. These issues complicate interpretation of results; for example, bacteriophages transferred alongside bacteria may severely impact the gut microbiota composition. A higher level of control could potentially be achieved through transplanting synthetic microbial communities. While this approach has been successfully implemented in diverse in vitro setups the complexity of microbial communities still hinders its efficient use as a routine scientific procedure in live animals.

2.2.3 Controlling the environment

In most laboratory studies, environmental complexity is reduced so that no, or very few, environmental parameters (usually only experimental treatments) vary among groups and subjects. Climate chambers and aquaria enable experiments by providing absolute control of abiotic conditions, such as light/dark cycles, humidity and temperature variations. Outdoor common garden experiments do not provide full control over environmental factors, but they ensure the effect on the systems being compared is identical. Some natural systems can also provide special conditions that enable environmental features to be controlled, such as cuckoo nestlings that are bred by other birds or salmon populations that breed in the same rivers in alternating years. Research on wild organisms usually incorporates more complex and dynamic environmental conditions. When controlling them is not possible, collection of relevant environmental metadata to be incorporated as covariates in the statistical analyses is useful. A century of ecological research has revealed the advantages of each of these approaches. On the one extreme, laboratory microcosms allow the most reductive control. On the other extreme, studies in the macrocosm of the real world provide perspective on emergent properties of natural ecosystems that cannot be anticipated solely based on microcosms.

2.3 Molecular resolution

The contents of this section have been extracted and modified from the article Disentangling host-microbiota complexity through hologenomics published in *Nature Reviews Genetics* in 2022 by the authors of the **Holo-omics Workbook**.

The complexity of a study system is not only determined by its inherent properties and study design, but also the techniques and procedures employed to analyse it. Researchers can decide how much a system is simplified by altering

the resolution of the hologenomic features under study; in essence, zooming in or zooming out.

2.3.1 Resolution of host genotypes

In host-microbiota studies, host genotypes can be defined at different levels, including species, breeds, populations, strains, sex or individuals. Genotypes can be defined as categorical variables, without analysing the differences between them, or can be studied in more detail through considering their actual genetic content and establishing correlations among them. When using an evolutionary perspective, phylogenetic relationships between genotypes are established based on phylogenomic markers, which usually vary above population and species level, but not among individuals. This implies that genomic variability among the individuals included within each genotype is overlooked. Studying the effect of interindividual genomic variability on host-microbiota systems, such as identifying candidate host genomic variants associated with microbial features, requires a higher level of resolution. This is achieved through defining genotypes at the individual level, and using techniques based on whole genome resequencing that enable the complexity of host genomes to be screened at a much finer level, so that differences between the individuals contrasted are not only defined based on their kinship, but also the functional properties of their genomic variants. Currently, this approach requires high quality reference genomes from which high density SNP profiles of individuals can be generated, for example through SNPchip or resequencing studies. The genomic resolution could be further refined by incorporating structural variants, methylation patterns, or even, we hypothesise, chromosome 3D folding structure as revealed through techniques such as Hi-C. In doing so, researchers can identify associations between SNPs or gene variants and specific microbiota traits, such as the relative abundance of certain taxa or the enrichment of a given function, and thus identify mechanisms by which a host exerts control over composition and function of its associated microbiota

Resolution of microbial metagenotypes

The structure and resolution at which microbial metagenotypes are defined also affects the complexity of the metagenome under analysis. Metagenotypes can be defined as arrays of microbial taxa, microbial genes or a combination of both. The most common approach to define them is to rely on short marker sequences targeted for metabarcoding purposes, such as the 16S rRNA or the internal transcribed spacer (ITS). However, these procedures often do not enable reliable taxonomic assignment at genus or species level, do not capture strain level community dynamics, and are prone to generate biased functional inferences, as bacteria with identical marker genes (particularly those associated with wild taxa) might carry very different catalogues of genes. Thus, while useful for estimating microbial diversity and obtaining preliminary insights into functionality, targeted sequencing approaches do not provide conclusive evidence about

the metabolic capabilities of the microbiota, particularly when working with non-human systems.

By contrast, if appropriate strategies and adequate sequencing depths are employed, shotgun metagenomics enables bacterial genome sequences to be recovered, from which genes can be predicted and annotated to create a gene catalogue that can define a metagenotype. However, these genes are not randomly distributed, but enclosed within genomes of specific bacteria or other microorganisms, with a particular combination of genes that shape their expression and the specific biological features (such as oxygen affinity, reproduction time, metabolic capacity) that determine their ecology. Hence, a more refined characterisation of microbial metagenotypes can be achieved through binning algorithms that enable bacterial genome reconstruction from metagenomic mixtures, yielding metagenome-assembled genomes (MAGs). Nevertheless, unless short-read sequencing is combined with long-read approaches, it is challenging to capture multi-copy genes such as the 16S rRNA marker gene 103, which is often employed in metabarcoding studies and therefore represents a useful link to a large number of existing studies. Machine learning-based solutions to link 16S rRNA marker gene sequences with MAGs are, however, being developed 104. Finally, regardless of the approach used to define the microbial metagenotype, the complexity of microbial communities will often require dimensionality reduction to increase statistical power 105,106. This can be achieved by defining co-abundance clusters, ecological guilds or more complex strategies that also consider temporal features of microbiota variation, such as compositional tensor factorisation.

Resolution of envirotypes

Characterisation of environmental factors that affect the host-microbiota system under study enable the definition of envirotypes, a term drawn from crop sciences that is useful for accounting for the environmental factors in the hologenomic context. Any different physical place, or place sampled at different time points, will be exposed to a different environment, as conditions will seldom be identical between two spatial and temporal points. Hence, the resolution at which the composite of environmental factors is considered will define whether these two environments will be considered different envirotypes or not. For example, if only considering water temperature, killer whales sampled in the Arctic and the Antarctic seas experience the same envirotype. However, if the biotic composition is also considered in the definition of the environment, the Arctic and the Antarctic will need to be split into two distinct envirotypes, as some killer whales will have access to penguins while others will not. The same principle applies to laboratory setups or mesocosm experiments: a temperature shift of 2-3 °C might not be considered relevant under some experimental setups, while it can define different envirotypes under other study designs. Finally, failure to recognise environmental factors that affect host-microbiota interactions, and thus define relevant envirotypes, can lead to increased noise and decreased

capacity to achieve statistical significance.

2.4 Spatiotemporal factors

The contents of this section have been extracted and modified from the article Disentangling host-microbiota complexity through hologenomics published in *Nature Reviews Genetics* in 2022 by the authors of the **Holo-omics Workbook**.

Spatial factors

Spatial resolution. Microbial communities associated with animal and plant hosts vary not only across coarse body parts, but also at the micro-scale, such as between the lumen and the intestinal crypts. Thus, the resolution at which a body site is defined will also determine how a hologenomic system is characterised. For example, the animal gastrointestinal tract can be considered a single sampling unit, 4-5 units or hundreds of micro-units, depending on the sampling and data processing strategies employed. Naturally, each level of resolution will allow different questions to be addressed and will require the use of different technologies and analytical approaches.

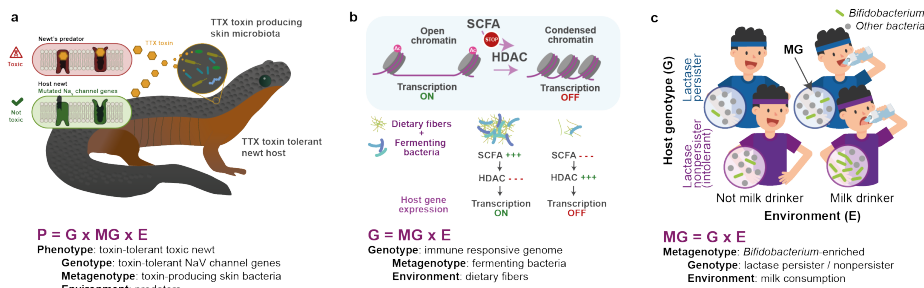
Temporal factors

Temporal features to be considered include when, how often, and for how long host-microbiota systems are to be analysed. When a host is first exposed to microbes with regard to temporal benchmarks (number of days or years) must be considered, as should the order in which it is exposed to them. Priority effects relate to how the order of species arrivals in an ecosystem shape the potential for subsequently arriving taxa to establish themselves. Although originally discussed at the macroorganismal level in the context of plant communities, the phenomenon is also relevant for building host-associated microorganism communities, for example as documented in the human gut. In addition, microbial communities are known to vary daily, seasonally and relative to life-stage patterns. Hence, the extent and frequency of sampling determine which of these dynamics will be observed or, conversely, missed. Finally, it is important to consider that the consequences of changes at one time period or life stage may appear only later in time, thus detection of such effects obviously requires that the subsequent period is also studied. For example, interventional animal experiments show that when the immune system develops early in life, there is a window of opportunity where the gut microbiota composition shapes the risk of developing diseases in the future.

2.5 Explanatory and response variables

The contents of this section have been extracted and modified from the article Disentangling host–microbiota complexity through hologenomics published in *Nature Reviews Genetics* in 2022 by the authors of the **Holo-omics Workbook**.

Host genomic and microbial metagenomic data generated under hologenomic setups can take on different roles when generating statistical models. While the environment is most often considered as an explanatory variable (though one can also study how the hologenome affects the environment), the host genome and the microbial metagenome are sometimes viewed as explanatory and sometimes as response variables, depending on the aim of the research. In many cases, directionality is set by the researcher rather than the biological system itself, as host-microbiota systems contain many bi-directional interactions and circular processes, which complicate the establishment of causal relationships. Here, we define three basic models in which the three main variables (genome, metagenome and environment) are assigned different roles to address different types of fundamental questions.



Examples of biological processes addressed by the different models of host-microbiota interactions. **a)** How does the hologenome shape animal phenotypes? Only the combination of specific host genomic (G) and microbial metagenomic (MG) features, probably developed due to a selective force exerted by the presence of predators (E) enables rough-skinned newts to have skin toxicity, an ecologically relevant phenotypic trait (P). **b)** How do the microbial metagenome and environment shape host genomic features? SCFA-producing bacteria along with a fibre-rich diet enhance chromatin accessibility and thus activate immune gene expression. **c)** How do the host genome and the environment shape microbial genomic features? Only the combination of a lactase nonpersister genotype combined with the milk-drinking envirotypes generates a microbial metagenotype characterised by enrichment of Bifidobacterium.

Phenotype as a product of genotype, metagenotype and enviotype

This is the main model used when hologenomics is conducted to ascertain how genome-metagenome-environment interactions affect the biological properties of a host, such as disease susceptibility, performance or fitness. It is an especially common and relevant model for health, agricultural, and ecological and evolutionary research 19,125–127. One clear example of a phenotype shaped by host genomic, microbial metagenomic and environmental factors was recently reported for rough-skinned newts. The study showed that bacteria on the skin of the newts produce a deadly neurotoxin from which the newt is protected by mutations in five host genes that encode the NaV channels normally targeted by the toxin. Thus, this ‘toxic newt’ phenotype is the result of both host and microbial genes, which likely evolved under the pressure exerted by an environmental factor, namely the presence of predators.

Genotype expression influenced by metagenotype and enviotype

When studying how core host genomic features, which contribute to shaping phenotypes, are affected by the microbiota, host genomic features become the response variable. Unlike the microbial metagenome, the genome sequence of the host organism is not variable, but microorganisms can induce chromatin remodelling and DNA methylation, and thus modulate the bioactivity of molecular receptors and host gene expression. A well-studied pathway that links the microbiota with host gene expression involves modulation of the activity of host histone deacetylases (HDAC) by short chain fatty acids (SCFA) produced by intestinal microorganisms. HDACs remove histone lysine acetyl groups, which leads to chromatin condensation and transcriptional silencing of genes. Increased SCFA concentrations inhibit histone deacetylases, thereby enhancing chromatin accessibility and activating gene expression. A metagenotype with a higher capacity to produce SCFAs combined with an enviotype characterised as a fibre-rich diet (required to produce SCFAs), therefore contributes to boost immune response through activating host immune gene expression.

Metagenotype as a product of genotype and enviotype

This model assumes the inverse causal directionality between the host genome and microbial metagenome to that described above. Candidate host genes related to microbiota features can be identified through GWAS in which the metagenotype (or derived metrics such as diversity or abundance of specific microbial taxa, genes or metabolic functions) are treated as a phenotypic trait. For instance, the increased abundance of lactose degrader *Bifidobacteria* in humans has been shown to be associated with lactase nonpersister genotype and consumption of milk (enviotype). Once candidate genes are known, targeted analyses in which natural or human-controlled genomic variability (such as the

number of copies of the amylase-encoding gene in humans) can be contrasted under controlled environmental conditions to ascertain the effect on metagenotypes (such as the abundance of Ruminococcaceae bacteria in the gut microbiota).

Part II

LABORATORY PROCEDURES

Chapter 3

About labwork

General considerations

Although the generation of each omic data layer requires dedicated protocols to be implemented, there are multiple general considerations that apply to all laboratory processes. In the following we list three of the most relevant ones.

External contamination

The risk of external contamination is a relevant issue that must be actively tackled when generating multi-omic data. External contamination refers to any molecule of interest that unintendedly is added to the sample, and analysed with the target molecules. As shotgun sequencing entails analysing all available nucleic acids in a sample, human saliva, skin microbiome, or microbes present in water and reagents are some of the sources of external contamination. Incorporating DNA and RNA from these sources can distort the biological signal, which can lead researchers into incorrect conclusions. The following measures contribute to minimise external contamination:

- Always wear gloves and work in sterile environments, such as clean laminar flow cabinets.
- Use filtered pipette tips.
- Separate pre-PCR and post-PCR laboratories.
- Process and sequence blank controls.

Internal cross-contamination

Another common type of contamination is that happening among samples. During the many pipetting actions laboratory protocols entail, is not uncommon to transfer small amounts of samples to adjacent tubes or wells. This can obviously distort the sample, and lead researchers into incorrect conclusions. The following measures contribute to minimise internal cross-contamination:

- Process all batches in an identical way for errors to be detectable.
- Avoid pipetting from the top of the tube to minimise sprays.
- Process and sequence blank controls.

Batch effects

The last global consideration is to be aware of batch effects and try to minimise or account for their impact in downstream analyses. Batch effects are almost unavoidable in holo-omic data generation, because samples are usually processed in batches. Each batch can suffer different types of technical biases, including the aforementioned contamination issues, but also other problems derived from the use of different reagent stocks, different researchers executing identical protocols in slightly different ways, or storing samples for variable time periods. The critical measure to minimise the impact of batch effects and account for them in downstream analysis is to randomise samples. Randomising means randomly assigning samples from different contrasting groups to the different batches, minimising correlation between batches and experimental groups. If this is done, laboratory batches can be included as covariates in statistical analyses, which enable accounting and controlling for batch effects in the final results.

Procedures for generating multi-omic data

This chapter contains sections dedicated to each of the omic layers included in the workbook.

- **Nucleic-acid sequencing-based approaches**
 - DNA/RNA extraction for **HG**, **HT**, **MG** and **MT**
 - Sequencing library preparation for **HG** and **MG**
 - Sequencing library preparation for **HT**
 - Sequencing library preparation for **MT**
- **Mass spectrometry-based approaches**
 - Protein extraction for **HP** and **MP**
 - Metabolite extraction for **ME**

Chapter 4

DNA/RNA extraction

Hundreds or (probably) thousands of different protocols and variations exist for extracting and purifying nucleic acids. Protocols can be classified based on methodological (e.g., chemical vs. physical DNA/RNA isolation, column-based vs bead-based, commercial vs. open-access).

Sample preprocessing

Bead-beating

Bead-beating is a mechanical disruption method performed before standard DNA extraction, where ceramic or glass beads are added to a tube containing microbial samples. Subsequently, moderate to high-speed shaking is applied to create collisions between the beads and samples. Bead-beating is widely used in microbial metagenomics studies for bacterial cell lysis, and various bead-beating protocols have been utilized to extract microbial DNA from stool samples. Literature has investigated the effects of different bead-beating techniques on downstream analyses [Zhang et al., 2021, Fiedorová et al. [2019]].

Freeze-heat shock

Temperature shocks are one of the most damaging processes for tissue, cell and DNA integrity. While such events are commonly avoided to preserve the quality of the samples, heat-shocks have been shown to improve nucleic acid extractions in various contexts. This is because freezing induces crystallisation of water inside cells which leads to destruction of cytoplasmic structures.

Tissue digestion

After tissue disaggregation, a typical approach involves treating samples with a detergent and salt (such as SDS) to rupture cell membranes and sometimes with

enzymes for cellular and organelle disruption and the elimination of impurities. Proteinase K is a popular choice for DNA isolation from mammalian tissues and cells, whereas lyticase and lysozyme are enzymes used to break down the cell walls of yeast and bacteria and are commonly featured in microbial DNA extraction kits.

Chemical isolation

Once the DNA is released, proteins and other contaminants must be removed. When using chemical approaches, this is typically done by adding a precipitating agent like alcohols (e.g., ethanol) or salts (e.g., ammonium acetate). This process separates DNA and contaminants in different phases, which enables the contaminants to be removed from the sample, thus purifying the DNA. Purely chemical procedures for DNA isolation are becoming less common for the challenges they entail for high-throughput sample processing and automation.

Physicochemical isolation

Physicochemical procedures are the most commonly employed strategies for DNA and RNA isolation. Two main strategies exist, either column-based or bead-based isolation.

Column-based

The spin column-based nucleic acid purification method is a rapid solid-phase extraction technique that purifies nucleic acids. The principle underlying this method is that under specific ionic conditions, nucleic acids bind to the solid-phase silica. A binding buffer is used to establish the optimal pH or salt concentration required for DNA to bind to silica. The sample in the binding buffer is then transferred to a spin column, which is placed in a centrifuge or attached to a vacuum. The centrifuge or vacuum forces the solution through a silica membrane within the spin column, where the nucleic acids bind to the silica membrane while the rest of the solution flows through. Once the target material is bound, the flow-through can be discarded.

The next step involves washing the column by adding a new buffer, which typically contains alcohol, to maintain binding conditions while removing binding salts and any remaining contaminants. This process requires several washes, often with increasing concentrations of ethanol or isopropanol, until the nucleic acids on the silica membrane are free of contaminants. Finally, elution is the process of adding an aqueous solution to the column, allowing the hydrophilic nucleic acid to leave the column and enter the solution. This step can be enhanced by altering the salt, pH, time, or temperature. Finally, to collect the extract, the column is transferred to a clean microtube prior to a final centrifugation step.

Bead-based

Bead-based nucleic acid extractions are based on the magnetic properties of very small (20 to 30 nm) iron oxide particles that only display magnetic behaviour in the presence of an external magnetic field. Several types of magnetic beads with different binding properties exist, which can be used for DNA and RNA purification as well as proteins and other biomolecules, depending on their surface coatings and chemistries. For instance, streptavidin-coated magnetic beads are commonly used for nucleic acid extractions, due to their capacity to bind biotinylated ligands such as nucleic acids.

First, beads are mixed with the sample along with a binding buffer that provokes DNA to get attached to the magnetic beads. Subsequently, particles (with attached DNA) are dragged to an edge of the tube by the magnetic force of an external magnet, thus immobilising them. While the beads are immobilised, the rest of the sample is removed. The bead-bound DNA is retained during the washing steps, and finally released to an aqueous solution when the sample has been purified. The main advantage offered by bead-based strategies is their capacity to upscale and automatise, because there is no need for vacuum or centrifugation.

Available protocols

Hundreds of protocols, both open access and commercial, are currently available to extract nucleic acids from different types of samples.

Contents of this section were created by Antton Alberdi.

Chapter 5

Protein/metabolite extraction

Laboratory protocols for protein/metabolite extraction

Chapter 6

Sequencing library preparation

Sequencing of DNA and RNA molecules require sequencing libraries to be prepared, which entails modifying original nucleic acid molecules to allow sequencing platforms to identify the target molecules and perform the sequencing.

Sequencing strategies and platforms

Library features are specific to each sequencing platform, which requires selecting in advance the sequencing strategy to be employed. Pure nucleic acid sequencing-based strategies can be broadly divided in two groups. Short-read sequencing (SRS) platforms provide large amounts of data yet with short sequencing reads (typically 150 nucleotides). In contrast, long-read sequencing (LRS) platforms yield much longer sequences (thousands or even million of nucleotides), yet with a lower throughput, and typically lower sequence quality. The SRS market is dominated by two main companies with proprietary platforms, namely Illumina and BGI, although PacBio recently released their own SRS platform called ONSO. The LRS market is also dominated by two different companies with proprietary technologies, which are Oxford Nanopore (ONT) and Pacific Biosciences (Pacbio).

Sequencing enterprises, as well as auxiliary biotechnological companies, provide library preparation kits that can be more or less customised for different purposes.

Technology	Platforms	Sequencing type	Company
Sequencing by synthesis (SBS)	MiSeq, NovaSeq	Short-read sequencing	Illumina

Technology	Platforms	Sequencing type	Company
Combinatorial probe-anchor synthesis (cPAS)	DNBSeq	Short-read sequencing	BGI
Sequencing by binding (SBB) technology	Onso	Short-read sequencing	PacBio
Single Molecule Real-Time sequencing (SMRT)	Sequel, Revio	Long-read sequencing	PacBio
Nanopore sequencing	MinION, GridION, PromethION	Long-read sequencing	Oxford Nanopore

Some of the most widely used sequencing technologies and platforms.

PCR-based vs. PCR-free library preparation

Sequencing library preparation procedures can be split into two main groups depending on whether they PCR-amplify or not the DNA templates. Unlike in the case of targeted amplicon sequencing, in which the objective is to amplify a specific target region, the aim of including a PCR step in shotgun-based library preparation is to increase the molarity of the library and/or to attach indices (see below) to the adaptors.

Learn more about PCR-based and PCR-free library preparation in this article by Jones et al. [Jones et al., 2015].

Indices and multiplexing

Usually, library preparation also entails tagging molecules with unique sample identifiers known as indices, which enable pooling molecules derived from multiple samples in a single sequencing run. This can be achieved in PCR-free protocols by using adaptors containing unique indices per sample, or by using indexed amplification primers in PCR-based library preparation protocols.

Learn more about indices and multiplexing in this article by Kircher et al. [Kircher et al., 2012].

Unique molecular identifiers (UMIs)

Unique molecular identifiers (UMIs) are a type of molecular barcoding that provides error correction and increased accuracy during sequencing by uniquely tag each molecule (rather than each pool of molecules derived from a sample)

in a sample library. UMIs are used for a wide range of sequencing applications, many around PCR duplicates in DNA and cDNA. UMI deduplication is also useful for RNA-seq gene expression analysis and other quantitative sequencing methods.

Learn more about unique molecular identifiers in this article by Kivioja et al. [Kivioja et al., 2011].

Contents of this section were created by Antton Alberdi.

6.1 Host genomics and microbial metagenomics

The library preparation strategies for generating host genomic (HG) and microbial metagenomic (MG) data are generally the same.

DNA fragmentation

Short-read sequencing libraries requires DNA to be sheared to the desired fragment-length (usually 400-500 nucleotides), which can be achieved using either chemical (e.g., restriction enzymes) or physical (e.g. ultrasonication) procedures. Some long-read sequencing libraries intend to keep the largest DNA molecules possible, although some others recommend fragmenting to optimal mid-length molecules (e.g., around 10,000 nucleotides for Pacbio HiFi). After fragmentation, many library preparation protocols require repairing molecule ends by converting 5'-protruding and/or 3'-protruding ends to 5'-phosphorylated, blunt-end (see below) molecules.

Adaptor ligation

In shotgun libraries adaptors are merged to DNA template molecules through chemical ligation (e.g., using a ligase enzyme). The ligation process is slightly different depending on whether the DNA template has blunt- or sticky-ends. In blunt ends, both strands are of equal length – i.e. they end at the same base position, leaving no unpaired bases on either strand, while in sticky ends, one strand is longer than the other. Some protocols deliberately create sticky-ends from blunt-end fragmented DNA molecules by adding a single adenine base to form an overhang by an A-tailing reaction. This A overhang allows adapters containing a single thymine overhanging base to pair with the DNA fragments.

An example of a blunt-end molecule:

```
5'-GATCTGACTGATGCGTATGCTAGT-3'
3'-CTAGACTGACTACGCATACGATCA-5'
```

An example of a sticky-end molecule:

```
5'-GATCTGACTGATGCGTATGCTAGT-3'
3'-CTAGACTGACTACGCATACGATC-5'
```

List of available protocols

Type	Name	Author/owner	Protocol/Article
SRS	Blunt-End Single-Tube (BEST) library prep protocol	Open access	Article
SRS	Santa Cruz Reaction (SCR) single-stranded library prep protocol	Open access	Article
SRS		Illumina	Protocol
LRS	SMRTbell prep kit 3.0 for PacBio HiFi Sequencing	Pacbio	Protocol

6.2 Host transcriptomics

Library preparation for host transcriptomics (HT) requires some extra steps to the already described procedures for host genomics and microbial metagenomics. This is due to two main reasons. First, because RNA molecules cannot be directly built into most sequencing libraries, and require instead to generate complementary DNA (cDNA) before library preparation. Second, because gene transcripts tend to be overwhelmingly dominated by rRNA and mtDNA genes, which are often not of interest for the researcher.

Sample quality assessment

Before starting any library preparation protocol assessing the quality of RNA samples is strongly recommended. While traditionally assessed through agarose gel electrophoresis, nowadays RNA quality assessment is performed on electropherogram profiles, which are produced by nucleic acid fragment analysis instruments (e.g. Bioanalyzer, Fragment Analyzer). Traditionally, a simple model evaluating the 28S to 18S rRNA ratio was used as a criterion for RNA quality. However, the most common metric currently employed for assessing the preservation quality of RNA is the RNA integrity number (RIN), which accounts for more RNA features for assessing sample quality [Schroeder et al., 2006]. RIN values range from 10 (intact RNA) to 1 (totally degraded RNA). For example, the poly(A) enrichment procedures explained below require high quality RNA (RIN > 8), because RNA degradation to breaks within the transcript body and due to the selection of the poly(A) tail, the 3' ends are enriched while the more 5' sequences would not be captured, leading to a strong 3' bias for degraded RNA inputs.

DNA removal

Depending on the RNA extraction method employed, it is not rare trace amounts of genomic DNA (gDNA) to be co-purified with RNA. Contaminating gDNA can interfere with reverse transcription and may lead to false positives, higher background, or lower detection in sensitive applications such as RT-qPCR. The traditional method of gDNA removal is the addition of DNase I to RNA extracts. DNase I must be removed prior to cDNA synthesis since any residual enzyme would degrade single-stranded DNA. Unfortunately, RNA loss or damage can occur during DNase I inactivation treatment. As an alternative to DNase I, double-strand-specific DNases are available to eliminate contaminating gDNA without affecting RNA or single-stranded DNAs.

Stranded vs. non-stranded transcriptomics

RNA-Seq libraries can be stranded or non-stranded (unstranded), a decision that affects data analysis and interpretation. Stranded RNA-Seq (also referred to as strand-specific or directional RNA-Seq) enables researchers to determine the orientation of the transcript, whereas this information is lost in non-stranded, or standard, RNA-Seq. Non-stranded RNA-Seq is often sufficient for measuring gene expression in organisms with well-annotated genomes, as with a reference transcriptome, it is possible to infer orientation for most of the sequencing reads. As there are fewer steps than stranded library preparation, the benefits of this approach are lower cost, simpler execution, and greater recovery of material, which renders non-stranded RNA-Seq the preferred option for holo-omic analyses. In contrast, stranded RNA-Seq is useful if the aims include annotating genomes, identifying antisense transcripts or discovering novel transcripts.

cDNA conversion

Most RNA-Seq experiments are carried out on instruments that sequence DNA molecules, rather than RNA. This implies that RNA conversion to cDNA is a required step before library preparation. The synthesis of cDNA from an RNA template is carried out via reverse transcription using reverse transcriptases. In nature, these enzymes convert the viral RNA genome into a complementary DNA (cDNA) molecule, which can integrate into the host's genome, among other processes.

Reverse transcription, similar to PCR, requires the use of primers. Two main types of primers:

- **Random primers:** this type of primers are oligonucleotides with random base sequences. They are often six nucleotides long and are usually referred to as random hexamers. While random primers help improve cDNA synthesis for detection, they are not suitable for full-length reverse transcription of long RNA. Increasing the concentration of random hexamers in reverse transcription reactions improves cDNA yield but results

in shorter cDNA fragments due to increased binding at multiple sites on the same template

- **oligo(dT) primers:** this type of primers consist of a stretch of 12–18 deoxythymidines that anneal to poly(A) tails of eukaryotic mRNAs (see the section below for further details).

Reverse transcription reactions for cDNA library construction and sequencing involve two main steps: first-strand cDNA synthesis and second-strand cDNA synthesis.

- **First-strand cDNA synthesis:** this initial step generates a cDNA:RNA hybrid through the below-described three-step process.
 - **Primer annealing:** in this step primers are attached to the RNA template, which usually happens before reverse transcriptase and necessary components (e.g., buffer, dNTPs, RNase inhibitor) are added.
 - **DNA polymerisation:** in this step the complementary DNA is polymerised by the reverse transcriptase enzyme. With oligo(dT) primers ($T_m \sim 35\text{--}50^\circ\text{C}$), the reaction is often incubated directly at the optimal temperature of the reverse transcriptase ($37\text{--}50^\circ\text{C}$), while random hexamers typically have lower T_m ($\sim 10\text{--}15^\circ\text{C}$) due to their shorter length. Using a thermostable reverse transcriptase allows, a higher reaction temperature (e.g., 50°C), to help denature RNA with high GC content or secondary structures without impacting enzyme activity. With such enzymes, high-temperature incubation can result in an increase in cDNA yield, length, and representation.
 - **Enzyme deactivation:** in this final step temperature is increased to $70\text{--}85^\circ\text{C}$, depending upon the thermostability of the enzyme, to deactivate the reverse transcriptase.
- **Second-strand cDNA synthesis:** in this second step the first-strand cDNA is used as a template to generate double-stranded cDNA representing the RNA targets. Synthesis of double-stranded cDNA often employs a different DNA polymerase to produce the complementary strand of the first cDNA strand.

rRNA depletion through poly-A enrichment

Ribosomal RNA (rRNA) helps translate the information in messenger RNA (mRNA) into protein. It is the predominant form of RNA found in most cells, which can make over 80% of cellular RNA despite never being translated into proteins itself. In consequence, most reads derived from RNA belong to rRNAs, unless depletion strategies are implemented.

Excessively abundant rRNA sequences can be depleted using multiple strategies, which are covered in the Microbial metatranscriptomics. The most broadly employed enrichment strategy when dealing with eukaryotic organisms is rRNA de-

pletion through poly-A enrichment. This strategy relies on the fact that mature coding mRNAs of eukaryotic organisms contain polyA tails, long chains (tens to hundreds) of adenine nucleotides that are added to primary RNA transcripts to increase the stability of the molecule. However, not all transcripts contain poly(A) tails. microRNAs, small nucleolar RNAs (snoRNAs), transfer RNAs (tRNAs), some long non-coding RNAs (lncRNAs), and even protein-coding mRNAs such as histone mRNAs do not contain poly(A) tails, thus will be removed together with rRNA during poly(A) selection. If interested in quantifying expression of such transcripts the use of alternative methods is recommended.

The most broadly employed strategies to deplete rRNA through poly-A enrichment rely either on hybridisation with Oligo(dT)-attached magnetic beads or oligo(dT) priming during cDNA conversion step. In the former strategy, poly(A)-containing RNA molecules hybridise with Oligo(dT) stretches attached to magnetic beads. Following hybridisation, the supernatant consisting of non-polyadenylated molecules is removed. The beads are washed prior to elution of the poly(A)-selected RNA in water or buffer.

List of available protocols

Type	Name	Author/owner	Protocol/Article
oligo(dT) hybridisation	Dynabeads Oligo (dT)25-61005	Thermo Fisher	Protocol
oligo(dT) priming			

Contents of this section were created by Antton Alberdi.

6.3 Microbial metatranscriptomics

Sequencing library preparation for microbial metatranscriptomics faces the same challenges as host genomics, but the fact that prokaryotic mRNA have no poly-A tails makes it impossible to apply oligo(dT)-based rRNA depletion strategies. There are three other alternatives through which prokaryotic rRNA can be depleted. These three strategies require designing oligos, probes or guides whose sequences complement the DNA sequences that should be removed. Most commercial kits contain probes designed to remove rRNA sequences of the most commonly employed animal hosts (Human/Mouse/Rat), as well as bacteria, but custom probes targeting any genes could be employed. The first two methods shown below are implemented before library-preparation, thus independent reactions must be ran for each sample. The last strategy is implemented after library preparation, which enables multiple indexed libraries to be pooled, and thus performing a single reaction per pool.

Capture-based rRNA depletion

This method relies on capture rRNA with complimentary oligos that are coupled to paramagnetic beads. Unwanted transcripts get bound to beads, which can then be retained using a magnet, while the non-hybridizing transcripts remain in the elution.

RNase-based rRNA depletion

A more recent technological upgrade to capture-based rRNA depletion is to, instead of using paramagnetic beads, degrade RNA:DNA hybrids using RNase H [Huang et al., 2020].

CRISPR/Cas9-based rRNA depletion

The newest method of all three relies on the DNA cleavage capacity of the Cas9 enzyme [Gu et al., 2016]. In this method, custom-designed guides are used for the Cas9 enzyme to cleave unwanted sequences. This strategy is applied once libraries are prepared, and before the final PCR amplification is conducted. When the targetted molecules are cleaved, they lack one of the two adaptors, and therefore they are not amplified, resulting in a considerable depletion compared to the rest of the library.

List of available protocols

Name	Strategy	Author/owner	Protocol/Article
Custom capture-based depletion	Capture-based	Open Source	Article [Kraus et al., 2019]
Legacy Ribo-Zero	Capture-based	Illumina	
Custom RNase-based depletion	RNase-based	Open Source	Article [Huang et al., 2020]
Ribo-Zero Plus	RNase-based	Illumina	
NEBNext® rRNA Depletion Kit	RNase-based	NEB	
DASH	Cas9-based	Open Source	Article [Prezza et al., 2020]

Contents of this section were created by Antton Alberdi.

Part III

BIOINFORMATIC PROCEDURES

Chapter 7

About bioinformatics

Bioinformatic processing of raw sequencing and mass spectrometry data is the computational step that precedes statistical analyses and integration of multi-omic data. Through bioinformatic processing raw data are converted into meaningful bits of information, usually drastically decreasing the size of the data sets that are used for downstream analyses.

Raw sequencing and mass spectrometry-based data files used in holo-omic analyses are typically in the realm of gigabytes (Gb) or even terabytes (Tb). Many of the performed operations require large amounts of memory (some more than 1Tb), which makes it impossible to process data in personal computers. Instead, most bioinformatics tasks are performed in computational clusters with access to large amounts of memory and many CPUs and GPUs, which enable parallelising computational tasks thus speeding up data processing time.

However, for the sake of simplicity and practicality, the example datasets included in this Workbook have been considerably downscaled to enable reproducing the exercises in personal computers.

All bioinformatic analyses included in the **Holo-omics workbook** are conducted in a Unix command line Shell environment (BASH/SH). You can find the details to set-up your SHELL environment in the section Prepare your Shell environment.

7.1 Prepare your shell environment

If the comment chunks of the code (text after `#`) is creating you problems, use the following code to disable interactive comments and avoid issues when copy-pasting code:

```
setopt interactivecomments
```

Required software

Bioinformatic pipelines for processing omic data require the use of dozens of software. All the required software are listed in the conda environment installation file available [here](#).

Install conda / miniconda

Conda is an open-source package management system and environment management system that quickly installs, runs, and updates packages and their dependencies. If **conda** is not installed in your system, the first step is to install **miniconda** (a free minimal installer for conda, enough to run the bioinformatic analyses explained in the workbook). Miniconda installers for Linux, Mac and Windows operating systems can be found in the following website: <https://docs.conda.io/en/latest/miniconda.html>

Once conda or miniconda is installed in your system, you should be able to create and manage your conda environments. You can test whether conda has been successfully installed using the following code:

```
conda -V
#> conda 22.11.1 #or whatever version you have installed
```

Install mamba (optional)

An optional step is to install **mamba**, which is a reimplementation of the conda package manager in C++, which speeds up many of the processes. Mamba can be installed through the command line using the **conda install** option.

```
conda install mamba -n base -c conda-forge
```

Create a conda environment

All the bioinformatic analyses explained in this workbook will be run within an environment containing all the necessary software. The file that specifies which software to install in the environment is available [here](#), and can be retrieved using **wget** (as shown in the code below), or downloading from the Internet browser. If using the latter option, don't forget to provide the absolute path to the 'holo-omics-env.yaml' file in the **mamba create** command.

```
wget https://raw.githubusercontent.com/holo-omics/holo-omics.github.io/main/bin/holo-omics-env.yaml
conda update conda #to ensure everything is updated
conda deactivate #deactivate any conda environment before creating a new one
conda env create -f holo-omics-env.yaml
rm holo-omics-env.yaml #remove installer file
```

As the environment contains dozens of softwares, the process of creating it will take a while. It is recommended to have a good Internet connection to speed-up

software download. Once the installation is over, you can double-check whether the environment has been successfully created using the following script:

```
conda activate holo-omics
#> (holo-omics) anttonalberdi@Anttons-MBP ~ %
```

The (holo-omics) specifies the environment you are at. To get out of the environment use:

```
conda env list
#> base * /Users/anttonalberdi/miniconda3
#> holo-omics /Users/anttonalberdi/miniconda3/envs/holo-omics
```

Activate the holo-omics conda environment

Whenever running the holo-omic analyses explained in this workbook, it will be necessary to activate the holo-omics environment through the following command:

```
conda activate holo-omics
#> (holo-omics) anttonalberdi@Anttons-MBP ~ %
```

Install software in conda environment

Content to be added.

#Example code goes here

7.2 Example data for bioinformatics

Contents to be added here.

#Example code goes here

7.3 Using snakemake for workflow management

Contents to be added here.

#Example code goes here

Chapter 8

Sequencing data preprocessing

The first step of the bioinformatic pipeline is to pre-process the raw sequencing data to prepare them for downstream analyses.

Preprocess the reads using fastp

Raw sequencing data require an initial preprocessing to get rid off low-quality nucleotides and reads, as well as any remains of sequencing adaptors that can mess around in the downstream analyses. An efficient way to do so is to use the software **fastp**, which can perform all above-mentioned operations in a single go and directly on compressed files.

fastP documentation can be found [here](#).

```
fastp \
  --in1 {input.r1i} --in2 {input.r2i} \
  --out1 {output.r1o} --out2 {output.r2o} \
  --trim_poly_g \
  --trim_poly_x \
  --low_complexity_filter \
  --n_base_limit 5 \
  --qualified_quality_phred 20 \
  --length_required 60 \
  --thread {threads} \
  --html {output.fastp_html} \
  --json {output.fastp_json} \
  --adapter_sequence {params.adapter1} \
  --adapter_sequence_r2 {params.adapter2}
```

Splitting host and non-host data

Depending on the sample type employed for data generation, sequencing data might contain only host reads, only microbial reads, or a mixture of both. For example, blood sampled from an animal is expected to only contain host DNA/RNA reads (unless an infection is ongoing), while DNA extracted from a microbial culture is only expected to contain microbial DNA/RNA reads (unless human contamination has happened). In contrast, intestinal content samples, faecal samples, leave samples or root samples can contain both host and microbial nucleic acids.

Index host genome

```
bowtie2-build \
  --large-index \
  --threads {threads} \
  {output.rn_catted_ref} {output.rn_catted_ref}
```

Map samples to host genomes

```
# Map reads to catted reference using Bowtie2
bowtie2 \
  --time \
  --threads {threads} \
  -x {input.catted_ref} \
  -1 {input.r1i} \
  -2 {input.r2i} \
| samtools view -b -@ {threads} - | samtools sort -@ {threads} -o {output.all_bam} - &&

# Extract non-host reads (note we're not compressing for nonpareil)
samtools view -b -f12 -@ {threads} {output.all_bam} \
| samtools fastq -@ {threads} -1 {output.non_host_r1} -2 {output.non_host_r2} - &&

# Send host reads to BAM
samtools view -b -F12 -@ {threads} {output.all_bam} \
| samtools sort -@ {threads} -o {output.host_bam} -
```

Chapter 9

Host genomics (HG) data processing

Contents to be added.

9.1 Host reference genome

Contents to be added.

9.2 Host genome resequencing

Contents to be added.

Chapter 10

Microbial metagenomics (MG) data processing

Microbial metagenomic data processing can be conducted following different strategies. Decision on which approach to use should be based on the aims of the study, available reference data, amount of generated data, and many other criteria. In this workbook we consider three main approaches that require different bioinformatic pipelines to be implemented.

- **Reference-based approach:** it relies on a reference database of microbial genomes to which sequencing reads can be mapped to obtain estimations of relative proportion of reads belonging to each of the genomes available in the reference database. It is the simplest and computationally less expensive approach, yet it completely relies on a complete and representative reference database.
- **Assembly-based approach:** it is based on assembling sequencing reads into longer DNA sequences known as contigs, which can then be used to predict genes and perform functional analyses. The main limitation of this approach is that the entire metagenome (set of contigs) in each sample is considered as a single unit, thus overlooking which bacterial genome each detected gene belongs to.
- **Genome-resolved approach:** it is the most advanced of the three approaches, and the strategy that provides the largest amount of information, as the aim of this approach is to directly reconstruct all the genomes in a metagenome. This is achieved by binning contigs into Metagenome-Assembled Genomes (MAGs), which can then be taxonomically and functionally annotated to perform sound community-level analyses.

Contents of this section were created by Antton Alberdi.

10.1 Reference-based

Reference-based metagenomics is the approach that aims at characterising metagenomes based on existing genome sequences that are used as reference. This approach is meaningful when the microbial community under study is well known, such as the human microbiome, or when a reference MAG catalogue has been generated from a subset of samples under analysis. When the genome catalogue used as a reference has not been generated from the same environment (e.g., using human gut microorganisms as reference for vulture gut microbiomes), there are two major risks. The first one is that some of the microorganisms present in the studied environment might not be represented in the reference catalogue, which results in diversity underestimations. The second one is that the microorganisms present in the reference catalogue are similar but not identical to the ones in the target sample, which might result in incorrect taxonomic and functional inferences. It is therefore important to keep these caveats in mind when performing reference-based metagenomic analyses, mainly when dealing with non-model host organisms.

10.2 Assembly-based

Assembly-based approaches for processing metagenomic data are based on assembling sequencing reads into longer DNA sequences known as contigs, which can then be used to predict genes and perform functional analyses. The main limitation of this approach is that the entire metagenome (set of contigs) in each sample is considered as a single unit, thus overlooking which bacterial genome each detected gene belongs to. Assembly-based approaches can be divided in two main strategies:

- Individual assembly-based
- Coassembly-based

Individual assembly-based

Two of the most popular metagenome assemblers are **Megahit** and **MetaSpades**. MetaSpades is considered superior in terms of assembly quality, yet memory requirements are much larger than those of Megahit. Thus, one of the most relevant criteria to choose the assembler to be employed is the balance between amount of data and available memory. Another minor, yet relevant difference between both assemblers is that Megahit allows removing contigs below a certain size, while MetaSpades needs to be piped with another software (e.g. `bbmap`) to get rid off barely informative yet often abundant short contigs.

Individual assembly using Megahit

```
megahit \
  -t {threads} \
```

```

--verbose \
--min-contig-len 1500 \
-1 {input.r1} -2 {input.r2} \
-o {params.workdir}
2> {log}

```

Individual assembly using MetaSpades

```

metaspades.py \
  -t {threads} \
  -k 21,33,55,77,99 \
  -1 {input.r1} -2 {input.r2} \
  -o {params.workdir}
2> {log}

# Remove contigs shorter than 1,500 bp using bbmap
reformat.sh \
  in={params.workdir}/scaffolds.fasta \
  out={output.assembly} \
  minlength=1500

```

Assembly statistics using Quast

The metagenome assemblies can have very different properties depending on the amount of data used for the assembly, the complexity of the microbial community, and other biological and technical aspects. It is therefore convenient to obtain some general statistics of the assemblies to decide whether they look meaningful to continue with downstream analyses. This can be easily done using the software **Quast**.

```

quast \
  -o {output.report} \
  --threads {threads} \
  {input.assembly}

```

Coassembly-based

Coassembly is the process of assembling input files consisting of reads from multiple samples, as opposed to performing an independent assembly for each sample, where the input would only include reads from that particular sample. Coassembly has several advantages, such as increased read depth, simplified comparison across samples by utilizing a single reference assembly for all, and frequently, a better capability to recover genomes from metagenomes by obtaining differential coverage information. However, it can also limit the capacity to recover strain-level variation.

Coassembling multiple samples does not require special assemblers, but only preparing the input files in the correct way to enable assemblers to perform the assembly over multiple samples. An example for metaspades is shown below:

```
#Concatenate input reads into a single big input file
cat {input.reads}/*_1.fq.gz > {params.r1_cat}
cat {input.reads}/*_2.fq.gz > {params.r2_cat}

# Run metaspades
metaspades.py \
  -t {threads} \
  -k 21,33,55,77,99 \
  -1 {params.r1_cat} -2 {params.r2_cat} \
  -o {params.workdir}
2> {log}

# Remove contigs shorter than 1,500 bp using bbmap
reformat.sh \
  in={params.workdir}/scaffolds.fasta \
  out={output.Coassembly} \
  minlength=1500
```

Note that the genome-resolved metagenomic approach also relies on assemblies or co-assemblies, but downstream binning procedures are explained in the **genome-resolved approach** section.

Gene annotation

Gene annotation refers to the process of identifying and assigning function to genes present in an assembly. In the first step, protein-coding and other types of genes are identified using tools such as Prodigal based on structural information of the DNA sequences. These software also predict the protein sequences these genes are expected to yield, which are then used to assign functions by contrasting them with functionally annotated reference databases. Due to the amount of reference databases available, it is common practice to match the genes against multiple databases and yield multiple annotations per gene. Currently, multiple tools exist that perform all these procedures in a single pipeline, such as DFAST [Tanizawa et al., 2017] and DRAM [Shaffer et al., 2020]. DFAST annotates genes against the TIGRFAM and Clusters of Orthologous Groups (COG) databases, while DRAM performs the annotation using Pfam, KEGG, UniProt, CAZY and MEROPS databases.

```
DRAM.py annotate \
  -i {input.assembly} \
  -o {outdir} \
  --threads {threads} \
  --min_contig_size 1500
```

The procedure for annotating MAGs, which is explained in the **genome-resolved approach** section, is identical to this one, with the only difference that the a MAG or a set of MAGs are used as input data rather than a metagenomic assembly.

Read mapping

The aim of assembly-based analyses is often to obtain gene-abundance information per studied sample, to characterise the functional properties of the entire metagenome as a whole (as opposed to the **genome-resolved approach** approach, in which functional attributes are assigned to each MAG). This requires reads from each sample to be mapped against the sequence of all protein-coding genes identified during the annotation process. All gene prediction software and annotation pipelines produce a FASTA file only containing gene sequences, which is used as the reference material.

The gene catalogue needs to be indexed before the mapping.

```
bowtie2-build \
  --large-index \
  --threads {threads} \
  {all_genes}.fa.gz
```

Then, the following step needs to be iterated for each sample, yielding a BAM mapping file for each sample.

```
bowtie2 \
  --time \
  --threads {threads} \
  -x {all_genes} \
  -1 {input.r1} \
  -2 {input.r2} \
  | samtools sort -@ {threads} -o {output}
```

Or relative abundance per gene per sample.

```
coverm genome \
  -b {params.BAMs}/*.bam \
  -s ^ \
  -m relative_abundance \
  -t {threads} \
  --min-covered-fraction 0 \
  > {output.mapping_rate}
```

Contents of this section were created by Antton Alberdi.

10.3 Genome-resolved

Genome-resolved metagenomics aims at recovering near-complete bacterial genomes from metagenomic mixtures. It relies on the assembling and read-mapping procedures explained in **assembly-based approach** section, which is followed by a binning procedure to produce the so-called Metagenome-Assembled Genomes (MAGs).

Note that entire suites and pipelines are available for conducting all the steps outlined in this section, and often more. Some of them include:

- Anvi'o
- metaWRAP
- ATLAS

Binning

Metagenomic binning is the bioinformatic process that attempts to group metagenomic sequences by their organism of origin {Goussarov}. In practice, what binning does is to cluster contigs of a metagenomic assembly into putative bacterial genomes. In the last decade over a dozen of binning algorithms have been released, each relying on different structural and mathematical properties of the input data.

Two of the most relevant structural properties to group contigs into bins are oligonucleotide composition of contigs and present of universally conserved genes in contigs. MaxBin, for example, relies on such universally conserved genes to initialize clusters, which are then expanded using the oligonucleotide composition of contigs. Besides structural attributes of contigs, the main quantitative measure used for binning is differential coverage, which is computed by counting the number of reads from different samples mapped to the assembly. This information is used by binning algorithms CONCOCT and MetaBat, for example.

Metabat and Maxbin require a depth file to be generated first.

```
jgi_summarize_bam_contig_depths \
  --outputDepth {output.depth} \
  {input.assemblybampath}
```

Example code for launching metabat2.

```
metabat2 \
  -i {input.assemblypath} \
  -a {input.depth} \
  -o {output.basepath} \
  -m 1500 \
  -t {threads} \
  --unbinned
```

Example code for launching MaxBin.

```
run_MaxBin.pl \
  -contig {input.assemblypath} \
  -abund {input.depth} \
  -out {output.basepath} \
  -thread {threads}
```

Bin refinement

The performance of the binning algorithms is largely dependent on the specific properties of each sample. A software that performs very well with a given sample can be easily outcompeted by another one in the next sample. In consequence, many researchers opt for ensemble approaches whereby assemblies are binned using multiple algorithms, followed by a refinement step that merges all generated information to yield consensus bins. This final step is often referred to as “bin refinement”, and can be performed using tools like metaWRAP [Uritskiy et al., 2018] or Dastool [Sieber et al., 2018]. Several benchmarking studies have shown that such ensemble approaches are usually better than individual binning tools.

The following code can be used to run an ensemble binning using metaWRAP.

```
metawrap binning -o {params.outdir} \
  -t {threads} \
  -m {params.memory} \
  -a {params.assembly} \
  -l 1500 \
  --metabat2 \
  --maxbin2 \
  --concoct \
```

The following code can be used to refine bins using metaWRAP.

```
metawrap bin_refinement \
  -m {params.memory} \
  -t {threads} \
  -o {params.outdir} \
  -A {params.concoct} \
  -B {params.maxbin2} \
  -C {params.metabat2} \
  -c 70 \
  -x 10
```

Bin quality assessment

Metagenomic binning is a powerful yet complex procedure that yields many bins that do not properly represent bacterial genomes. It is therefore essential to assess the quality of those bins before considering them representative

of bacterial genomes. The two main parameters used for bin assessment are completeness and contamination. Completeness refers to the fraction of a given bacterial genome estimated to be represented in the bin, while contamination refers to the proportion of the bin estimated to belong to a different genome. The most commonly employed software to assess bin quality is CheckM, which yields completeness and contamination metrics based on single-copy core genes.

Based on completeness and contamination metrics, a group of experts proposed some community standards to classify bins according to their quality and establish minimum quality requirements for considering a bin as a MAG [Bowers et al., 2017].

Bin curation

Contamination is an issue that in certain cases can be minimised by curating bins. The Anvi'o suite [Eren et al., 2015] provides a powerful visual interface to manually curate bins by dropping contigs that display distinct features (e.g., taxonomic annotation, coverage, GC%) to the rest of the contigs included in a bin. GUNC provides a way to implement a similar curation step in a more automatised manner [Orakov et al., 2021].

Dereplication

Dereplication is the reduction of a set of MAGs based on high sequence similarity between them [Evans and Deneff, 2020]. Although this step is neither essential nor meaningful in certain cases (e.g., when studying strain-level variation or pangenomes), in most cases it contributes to overcome issues such as excessive computational demands, inflated diversity or inspecific read mapping. If the catalogue of MAGs used to map sequencing reads to (see read mapping section below) contains many similar genomes, read mapping results in multiple high-quality alignments. Depending on the software used and parameters chosen, this leads to sequencing reads either being randomly distributed across the redundant genomes or being reported at all redundant locations. This can bias quantitative estimations of relative representation of each MAG in a given metagenomic sample.

Dereplication is based on pairwise comparisons of average nucleotide identity (ANI) between MAGs. This implies that the number of comparisons scales quadratically with an increasing amount of MAGs, which requires for efficient strategies to perform dereplication in a cost-efficient way. A popular tool used for dereplicating MAGs is dRep [Olm et al., 2017], which combines the fast yet inaccurate algorithm MASH with the slow but accurate gANI computation to yield a fast and accurate estimation of ANIs between MAGs. An optimal threshold that balances between retaining genome diversity while minimising cross-mapping issues has been found to be 98% ANI.

Taxonomic annotation

Although not necessary for conducting most of the downstream analyses, taxonomic annotation of MAGs is an important step to provide context, improve comparability and facilitate result interpretation in holo-omic studies. MAGs can be taxonomically annotated using different algorithms and reference databases, but the Genome Taxonomy Database (GTDB) [Parks et al., 2022] and associated taxonomic classification toolkit (GTDB-Tk) [Chaumeil et al., 2022] have become the preferred option for many researchers.

Functional annotation

Functional annotation refers to the process of identifying putative functions of genes present in MAGs based on information available in reference databases. As explained in the **assembly-based approach**, the first step is to predict genes in the MAGs (unless these are available from the assembly), followed by functional annotation by matching the protein sequences predicted from the genes with reference databases. Currently, multiple tools exist that perform all these procedures in a single pipeline, such as DFAST [Tanizawa et al., 2017] and DRAM [Shaffer et al., 2020]. DFAST annotates genes against the TIGRFAM and Clusters of Orthologous Groups (COG) databases, while DRAM performs the annotation using Pfam, KEGG, UniProt, CAZY and MEROPS databases.

```
DRAM.py annotate \
  -i {input.MAG} \
  -o {outdir} \
  --threads {threads} \
  --min_contig_size 1500
```

These functional annotations can be used for performing functional gene enrichment analyses, distilling them into genome-inferred functional traits, and many other downstream operations explained in the statistics part.

Read mapping

When the objective of a genome-resolved metagenomic analysis is to reconstruct and analyse a microbiome, researchers usually require relative abundance information to measure how abundant or rare each bacteria was in the analysed sample. In order to achieve this, it is necessary to map the reads of each sample back to the MAG catalogue and retrieve mapping statistics. The procedure is identical to that explained in the assembly read-mapping section, yet using the MAG catalogue as a reference database rather than the metagenomic assembly. This procedure usually happens in two steps. In the first step, reads are mapped to the MAG catalogue to generate BAM or CRAM mapping files. In the second step, these mapping files are used to extract quantitative read-abundance information in the form of a table in which the amount of reads mapped to each MAG in each sample is displayed.

First, all MAGs need to be concatenated into a single file, which will become the reference MAG catalogue or database.

```
cat {MAG.path}/*.fa.gz > {all_MAGs}.fa.gz
```

The MAG catalogue needs to be indexed before the mapping.

```
bowtie2-build \
  --large-index \
  --threads {threads} \
  {all_MAGs}.fa.gz
```

Then, the following step needs to be iterated for each sample, yielding a BAM mapping file for each sample.

```
bowtie2 \
  --time \
  --threads {threads} \
  -x {all_MAGs} \
  -1 {input.r1} \
  -2 {input.r2} \
  | samtools sort -@ {threads} -o {output}
```

Finally, CoverM can be used to extract the required stats, such as covered fraction per MAG per sample.

```
coverm genome \
  -b {input} \
  -s ^ \
  -m count covered_fraction length \
  -t {threads} \
  --min-covered-fraction 0 \
  > {output.count_table}
```

Or relative abundance per MAG per sample.

```
coverm genome \
  -b {params.BAMs}/*.bam \
  -s ^ \
  -m relative_abundance \
  -t {threads} \
  --min-covered-fraction 0 \
  > {output.mapping_rate}
```

Contents of this section were created by Antton Alberdi.

Chapter 11

Host transcriptomics (HT) data processing

The analysis of host transcriptomic data can be conducted following two main strategies, which depend on whether a well-annotated reference genome that contain all gene sequences of the studied transcripts is available or not. This is seldom the case in non-model organisms that lack complete reference genomes with high-quality annotation of genetic information, although many reference genome generation initiatives are rapidly increasing the number of available reference genomes. The pros and cons of using either approach have been addressed in the literature [Lee et al., 2021].

In the following two chapters, you will find example pipelines to process host transcriptomic data through both strategies:

- **Reference-based host transcriptomics (HT) data processing**
- **Reference-free host transcriptomics (HT) data processing**

11.1 Reference-based host transcriptomics (HT) data processing

Quality-filtering

Text to be added.

```
fastp \
  --in1 {input.read1} --in2 {input.read2} \
  --out1 {output.read1} --out2 {output.read2} \
  --trim_poly_g \
  --trim_poly_x \
```

```

--low_complexity_filter \
--n_base_limit 5 \
--qualified_quality_phred 20 \
--length_required 60 \
--thread {threads} \
--html {output.fastp_html} \
--json {output.fastp_json} \
--adapter_sequence AGATCGGAAGAGCACACGTCTGAACTCCAGTCA \
--adapter_sequence_r2 AGATCGGAAGAGCGTCGTGTAGGGAAAGAGTGT

```

Ribosomal RNA removal

Ribodetector can be used for efficient removal of rRNA sequences from host transcriptomics data, improving accuracy and reducing computational time. This tool helps to better identify transcripts and understand gene expression in complex microbiomes.

```

ribodetector_cpu \
-t 24 \
-l 150 \
-i {input.r1} {input.r2} \
-e rrna \
-o {output.non_rna_r1} {output.non_rna_r2}

```

Reference genome indexing

In order to use STAR for host transcriptomics, it is necessary to first generate a genome index, which can be used for multiple RNA-seq experiments.

```

STAR \
--runMode genomeGenerate \
--runThreadN {threads} \
--genomeDir {input} \
--genomeFastaFiles {input}/*.fna \
--sjdbGTFfile {input}/*.gtf \
--sjdbOverhang {params.readlength}

```

Text to be added.

Read mapping against reference genome

STAR (Spliced Transcripts Alignment to a Reference) is a fast and efficient method for aligning RNA-seq reads to a reference genome. It uses a two-pass alignment approach to detect spliced transcripts, improve accuracy and speed up the alignment process.

```
STAR \
  --runMode alignReads \
  --runThreadN {threads} \
  --genomeDir {params.genome} \
  --readFilesIn {input.read1} {input.read2} \
  --outFileNamePrefix {wildcards.sample} \
  --outSAMtype BAM SortedByCoordinate \
  --outReadsUnmapped Fastx \
  --readFilesCommand zcat \
  --quantMode GeneCounts
```

Contents of this section were created by Antton Alberdi and Raphael Eisenhofer.

11.2 Reference-free host transcriptomics (HT) data processing

Text to be added.

Chapter 12

Microbial metatranscriptomics (HT) data processing

The analysis of microbial metatranscriptomic data can be conducted following two main strategies, which depend on whether a reference catalogue of annotated bacterial genomes is available or not.

In the following two chapters, you will find example pipelines to process microbial metatranscriptomic data through both strategies:

- **Reference-based microbial metatranscriptomics (MT) data processing**
- **Reference-free microbial metatranscriptomics (MT) data processing**

12.1 Reference-based microbial metatranscriptomics (MT) data processing

Introduction to be added here.

Quality filtering

Fastp is a high-performance FASTQ preprocessor that can be used to clean up raw sequencing reads from Illumina platforms. It provides various quality control, filtering, and trimming options to remove low-quality bases, contaminants, and adapter sequences. The code provided performs a number of these steps, including trimming of poly-G and poly-X tails, which are commonly observed

in Illumina reads, filtering reads based on quality and length, and removing adapter sequences. The resulting cleaned reads are then written to the specified output files. Additionally, fastp provides comprehensive quality control reports in both HTML and JSON formats, which can be used to assess the quality of the input reads and the impact of the processing steps. Overall, pre-processing raw sequencing reads with fastp is a critical step in ensuring the accuracy and reliability of downstream bioinformatics analyses.

```
fastp \
  --in1 {input.r1i} --in2 {input.r2i} \
  --out1 {output.r1o} --out2 {output.r2o} \
  --trim_poly_g \
  --trim_poly_x \
  --n_base_limit 5 \
  --qualified_quality_phred 20 \
  --length_required 60 \
  --thread {threads} \
  --html {output.fastp_html} \
  --json {output.fastp_json} \
  --adapter_sequence CTGTCTCTTATACACATCT \
  --adapter_sequence_r2 CTGTCTCTTATACACATCT
```

Ribosomal RNA removal

Ribodetector can be used for efficient removal of rRNA sequences from microbial metatranscriptomics data, improving accuracy and reducing computational time. This tool helps to better identify transcripts and understand gene expression in complex microbiomes.

#Code to be added here

Host genome indexing

In order to use STAR for host transcriptomics, it is necessary to first generate a genome index, which can be used for multiple RNA-seq experiments.

#Code to be added here

Host genome mapping

STAR (Spliced Transcripts Alignment to a Reference) can be used for efficiently mapping host reads against a selected reference genome, and thus filter them out from subsequent metatranscriptomic analyses.

#Code to be added here

Generating and indexing the microbial genome catalogue

Explanations to be added here.

```
#Code to be added here
```

Mapping against the microbial genome catalogue

Explanations to be added here.

```
#Code to be added here
```

Calculate gene counts

Explanations to be added here.

```
#Code to be added here
```

12.2 Reference-free microbial metatranscriptomics (MT) data processing

Contents to be added here.

Chapter 13

Host proteomics (HP) data processing

Contents to be added.

Chapter 14

Microbial metaproteomics (MP) data processing

Contents to be added.

Part IV

STATISTICAL PROCEDURES

Chapter 15

About statistics

Statistics is probably the most challenging step of holo-omic studies, due to two main factors: the extreme complexity of the data, often containing thousands of features, and the limited sample size, often in the realm of the dozens of sampling units. This combination renders many holo-omic datasets rather statistics unfriendly.

A step-by-step approach

In this workbook we strongly encourage researchers to proceed step-by-step when dealing with holo-omics data and biological questions.

Initial quantitative exploration of omic layers

The analysis of any multi-omic data should begin with independent analysis of each omic layer to learn about its structure and variability before jumping to multi-omic data integration.

- **Data transformations:** multivariate datasets consist of different data types (e.g., presence-absence of taxa, counts of genes, community-level metabolic capacity index of a function, concentrations of metabolites across samples) that may require specific transformation before applying statistical techniques.
- **Unsupervised exploration of omic layers:** include exploratory techniques, such as cluster analysis and ordination-based visualisation methods, which reveal the structure and main patterns of the omic datasets without prior information about experimental design. These procedures might reveal that the observations are structured into meaningful groups or that variables can be reduced to fewer dimensions.
- **Supervised analysis of omic layers:** this type of analyses incorporate information of experimental design and aim at testing and estimating the

effects of the experimental factors (e.g., dietary treatment, drug administration) or variables of interest (e.g., age of the experimental subjects, geographic location of studied populations) on different omic layers.

Multi-omic data integration

When it comes to multi-omic data integration, the approaches can be broadly categorised into two types: multi-staged analysis and meta-dimensional or simultaneous analysis.

- **Multi-staged integration:** leverages the central dogma of molecular biology to assume that the variation in omic datasets is hierarchical, such that variation in DNA leads to variation in RNA and so on to determine the phenotype
- **Meta-dimensional integration:** considers the possibility that the phenotype is the product of the combination of variation across all omic layers, with the presence of complex inter-omic interactions.

All statistical analyses included in the **Holo-omics workbook** are conducted in R environment. You can find the details to set-up your R environment in the section Prepare your R environment.

15.1 Prepare your R environment

All statistical analyses included in the **Holo-omics workbook** are conducted in R environment [R Development Core Team, 2008]. R is a free software environment for statistical computing and graphics. It compiles and runs on a wide variety of UNIX platforms, Windows and MacOS, and in order to use it, R or RStudio must be installed in your local computer or remote server.

Required packages

In order to reproduce the analyses shown in the workbook, a rather long list of R packages must be installed. Packages are the fundamental units of reproducible R code, which include reusable R functions, the documentation that describes how to use them, and sample data.

- ape
- DESeq2
- distillR
- ggplot2
- tidyverse
- vegan
- (...)

Package installation

Packages are installed programatically using three main ways: through CRAN, Bioconductor or Github.

Install package from CRAN

CRAN is a network of ftp and web servers around the world that store identical, up-to-date, versions of code and documentation for R. Packages stored in CRAN can be installed using the following code:

```
install.packages("package_name")  
#e.g.  
install.packages("vegan")
```

Install package from Bioconductor

Bioconductor is a free, open source and open development software project for the analysis and comprehension of genomic data generated by wet lab experiments in molecular biology. Packages included in Bioconductor can be installed using the following code:

```
if (!require("BiocManager", quietly = TRUE))  
  install.packages("BiocManager")  
BiocManager::install("package_name")  
#e.g.  
BiocManager::install("DESeq2")
```

Install package from Github

GitHub is a code hosting platform for version control and collaboration. Packages stored in R can be installed using the following code after installing the package devtools:

```
library(devtools)  
install_github("github_repository_name_of_the_package")  
#e.g.  
install_github("anttonalberdi/distillR")
```

15.2 Example data for statistics

Contents to be added here.

Chapter 16

Single omic analyses

Multi-omic data analysis should start by evaluating each individual layer separately to gain insight into its structure and variability, before combining all layers. Despite the varying nature of the seven omic layers discussed in this guidebook, they all possess the common attribute of being multivariate, meaning they consist of multiple features, such as genomic variants, genes, metabolic pathways, proteins or metabolites, collected from multiple observations. This section contains the following three chapters:

- **Data transformations:** multivariate datasets consist of different data types (e.g., presence-absence of taxa, counts of genes, community-level metabolic capacity index of a function, concentrations of metabolites across samples) that may require specific transformation before applying statistical techniques.
- **Unsupervised exploration of omic layers:** include exploratory techniques, such as cluster analysis and ordination-based visualisation methods, which reveal the structure and main patterns of the omic datasets without prior information about experimental design. These procedures might reveal that the observations are structured into meaningful groups or that variables can be reduced to fewer dimensions.
- **Supervised analysis of omic layers:** this type of analyses incorporate information of experimental design and aim at testing and estimating the effects of the experimental factors (e.g., dietary treatment, drug administration) or variables of interest (e.g., age of the experimental subjects, geographic location of studied populations) on different omic layers.

Chapter 17

Data transformations

Before conducting any analysis on multivariate datasets, it is important to note that the different data types present (such as the presence/absence of taxa, gene counts, metabolic capacity indices, and metabolite concentrations) may need to undergo specific transformations.

17.1 Transformations to normalise data

Many statistical techniques used for such datasets have assumptions, such as normally (or symmetrically) distributed values, independence between variables, and homogeneity in population variances. Unfortunately, biological datasets often do not meet these requirements. However, through transformations, the original values can be modified to better conform to these assumptions.

#Example code to be added here

17.2 Transformations to account for missing observations

Many standard statistical techniques based on Euclidean distances, such as PCA, RDA, and K-means clustering, are not appropriate for analysing biological datasets that frequently contain missing observations represented as zeros. It is therefore recommended to use alternative transformations, such as Hellinger, when utilising these methods on datasets characterised by zero-inflated presence, absence, and abundance.

#Example code to be added here

17.3 Transformations to account for compositional nature of the data

When dealing with multi-omic datasets, it's important to determine if the measurements represent absolute or relative values. While HG provides qualitative information about host genomes, MG, HT, and MT provide quantitative information that's dependent on the amount of sequencing performed and thus are compositional. Consequently, raw quantitative values of genome abundance or gene expression across samples cannot be compared directly. To address this, the most common solution is to transform the raw abundance values into relative abundance data for comparison. However, this transformation reduces the independence of individual variables, which is an assumption for many statistical methods. An alternative is using ratio transformations like the centred log-ratio, which better suit compositional data analysis by removing the effect of the constant-sum constraint on the covariance and correlation matrices.

#Example code to be added here

17.4 Transformations to account for scaling

It's important to keep in mind that scaling can also impact the results when analyzing multi-omic data. For instance, in ME data, certain metabolites such as ATP may have much higher concentrations than other important metabolites like signaling molecules, potentially overshadowing significant differences in the less abundant yet meaningful metabolites. In Transcriptomics, transcript length biases may also cause similar distortions. One solution is to standardise the features by using transformations like z-score normalisation, but this can amplify the influence of measurement error that is typically higher for less abundant features.

#Example code to be added here

Chapter 18

Unsupervised exploration

Unsupervised methods in multi-omic data analysis involve techniques for exploring the structure and patterns of the data without prior knowledge of the experimental design. These include cluster analysis and visualisation methods based on ordination. These procedures can reveal meaningful groupings among observations or allow for reducing the complexity of the data by reducing the number of dimensions. Researchers can then use the results from these exploratory techniques in further multi-omic data integration. It's crucial to properly pre-process the data and choose the appropriate association coefficient when computing these methods, as this has a significant impact on the final outcome.

- Cluster analysis
- Dimension reduction and ordination

18.1 Cluster analysis

Clustering procedures group features or observations into homogeneous sets by minimising within-group and maximising among-group distances

18.1.1 Hierarchical clustering

Hierarchical clustering produces a stratified organisation of features or observations where relatively similar objects are grouped together. The clustering can be performed using different criteria to measure the distance between clusters, which will affect the final outcome of the analysis (e.g., single linkage, complete linkage, average linkage and Ward's minimum variance).

#Example code goes here

A useful exploratory analysis to reveal general patterns in an omic layer can be

obtained by simultaneous application of hierarchical clustering to the rows and columns of the data matrix, and visualising the results in a heatmap.

`#Example code goes here`

18.1.2 Disjoint clustering

Disjoint clustering techniques aim at separating the objects into individual, usually mutually exclusive, and in most cases, unconnected clusters. K-means clustering is one of the most typical algorithms where objects are assigned to k clusters using an iterative procedure that minimises the within-clusters sums of squares. Other available clustering methods include twinspace, self-organising maps, dbscan and Dirichlet multinomial mixtures (DMM). DMM were specifically developed to analyse MG data but can be equally useful for other sequencing-based omic datasets.

`#Example code goes here`

18.2 Dimension reduction and ordination

Ordination is a method complementary to data clustering, which enables displaying differences among samples graphically through reducing the dimensions of the original data set, so that similar objects are near and dissimilar objects are farther from each other.

18.2.1 Principal Component Analysis (PCA)

Principal component analysis (PCA) is one of the most widely applied methods for ordination. PCA generates new synthetic variables (principal components) that are linear combinations of the original variables and capture as much variance of the original data as possible. The principal components are orthogonal to each other and correspond to the successive dimensions of maximum variance of the scatter of points. The distance preserved among objects is euclidean and the relationships among variables are linear, thus PCA should generally be applied after appropriate transformations.

`#Example code goes here`

18.2.2 Principal Coordinate Analysis (PCoA)

Content to be added here.

`#Example code goes here`

18.2.3 Non-metric Multidimensional Scaling (NMDS)

Content to be added here.

#Example code goes here

18.2.4 t-Distributed Stochastic Neighbour Embedding (t-SNE)

Content to be added here. Requires many data points.

#Example code goes here

18.2.5 Uniform manifold approximation and projection (UMAP)

Content to be added here.

#Example code goes here

18.2.6 Potential of heat diffusion for affinity-based transition embedding (PHATE)

Content to be added here.

#Example code goes here

Chapter 19

Supervised analysis

The difference between supervised and unsupervised analyses in omic studies lies in the incorporation of experimental design information. Supervised analyses can be divided into two types: regression and classification. Regression problems involve predicting a numeric variable or matrix based on the omic data and experimental factors, such as treatment or subject characteristics. Classification problems, on the other hand, involve classifying observations into groups based on their features across different omic layers. Unlike unsupervised methods, supervised analyses incorporate prior information about the experimental design, making them useful for testing the effects of experimental factors and associating omic data with phenotypic features.

19.1 Regression methods

Independently testing the effects of the experimental factors of interest on different omic layers can be very informative to get an overall picture of how the host and the microbiome are responding to the environment and/or the experimental treatment.

19.1.1 PERMANOVA

Content to be added here.

```
#Example code goes here
```

19.1.2 ANOSIM

Content to be added here.

```
#Example code goes here
```

19.1.3 Redundancy analysis (RDA)

Content to be added here.

`#Example code goes here`

19.1.4 Canonical Correspondence Analysis (CCA)

Content to be added here.

`#Example code goes here`

19.1.5 Generalised linear modelling (GLM)

Content to be added here.

`#Example code goes here`

19.1.6 Generalised linear mixed modelling (GLMM)

Content to be added here.

`#Example code goes here`

19.2 Classification methods

It is in classification problems where ML algorithms have proven most useful.

19.2.1 Random Forests (RF)

Content to be added here.

`#Example code goes here`

19.2.2 Support Vector Machines (SVM)

Content to be added here.

`#Example code goes here`

Chapter 20

Multi-omic integration

Here is a review of existing methods.

Chapter 21

Multi-staged integration

Here is a review of existing methods.

Chapter 22

Meta-dimensional integration

Here is a review of existing methods.

Part V

RESOURCES

Chapter 23

Useful links

Genomics

- **Data Wrangling and Processing for Genomics (website):**

Shell command line usage

- **Introduction to the Command Line for Genomics (website):** general overview of basic command line usage.

R usage (General usage and programming)

- **Intro to R and RStudio for Genomics (website):**
- **Efficient R programming (website):** best practices for programming in R.

R usage (Graphics and visualisation)

- **Fundamentals of Data Visualization (website):** guide to making visualisations that accurately reflect the data, tell a story, and look professional.
- **R Graphics Cookbook (website):** a practical guide that provides more than 150 recipes to generate high-quality graphs using ggplot2.

Statistics

- **An Introduction to Statistical Learning (book):** freely available book about general statistical learning covering regression and classification problems through linear modelling and machine learning.
- **High dimensional statistics with R (website):** virtual lesson specialised in dealing with high dimensional data.

Chapter 24

References

Bibliography

Antton Alberdi, Sandra B Andersen, Morten T Limborg, Robert R Dunn, and M Thomas P Gilbert. Disentangling host-microbiota complexity through hologenomics. *Nat. Rev. Genet.*, 23(5):281–297, May 2022.

Robert M Bowers, Nikos C Kyrpides, Ramunas Stepanauskas, Miranda Harmon-Smith, Devin Doud, T B K Reddy, Frederik Schulz, Jessica Jarett, Adam R Rivers, Emiley A Eloë-Fadrosh, Susannah G Tringe, Natalia N Ivanova, Alex Copeland, Alicia Clum, Eric D Becraft, Rex R Malmstrom, Bruce Birren, Mircea Podar, Peer Bork, George M Weinstock, George M Garrity, Jeremy A Dodsworth, Shibu Yooseph, Granger Sutton, Frank O Glöckner, Jack A Gilbert, William C Nelson, Steven J Hallam, Sean P Jungbluth, Thijs J G Ettema, Scott Tighe, Konstantinos T Konstantinidis, Wen-Tso Liu, Brett J Baker, Thomas Rattei, Jonathan A Eisen, Brian Hedlund, Katherine D McMahon, Noah Fierer, Rob Knight, Rob Finn, Guy Cochrane, Ilene Karsch-Mizrachi, Gene W Tyson, Christian Rinke, Genome Standards Consortium, Alla Lapidus, Folker Meyer, Pelin Yilmaz, Donovan H Parks, A M Eren, Lynn Schriml, Jillian F Banfield, Philip Hugenholtz, and Tanja Woyke. Minimum information about a single amplified genome (MISAG) and a metagenome-assembled genome (MIMAG) of bacteria and archaea. *Nat. Biotechnol.*, 35(8):725–731, August 2017.

Pierre-Alain Chaumeil, Aaron J Mussig, Philip Hugenholtz, and Donovan H Parks. GTDB-Tk v2: memory friendly classification with the genome taxonomy database. *Bioinformatics*, 38(23):5315–5316, November 2022.

A Murat Eren, Özcan C Esen, Christopher Quince, Joseph H Vineis, Hilary G Morrison, Mitchell L Sogin, and Tom O Delmont. Anvi'o: an advanced analysis and visualization platform for 'omics data. *PeerJ*, 3:e1319, October 2015.

Jacob T Evans and Vincent J Deneff. To dereplicate or not to dereplicate? *mSphere*, 5(3), May 2020.

Kristýna Fiedorová, Matěj Radvanský, Eva Němcová, Hana Grombířková, Juraj Bosák, Michaela Černochová, Matej Lexa, David Šmajš, and Tomáš Freiberger. The impact of DNA extraction methods on stool bacterial and fungal microbiota community recovery. *Front. Microbiol.*, 10:821, April 2019.

- Caleb N Fischer, Eric P Trautman, Jason M Crawford, Eric V Stabb, Jo Handelsman, and Nichole A Broderick. Metabolite exchange between microbiome members produces compounds that influence drosophila behavior. *Elife*, 6, January 2017.
- W Gu, E D Crawford, B D O'Donovan, M R Wilson, E D Chow, H Retallack, and J L DeRisi. Depletion of abundant sequences by hybridization (DASH): using cas9 to remove unwanted high-abundance species in sequencing libraries and molecular counting applications. *Genome Biol.*, 17:41, March 2016.
- Yiming Huang, Ravi U Sheth, Andrew Kaufman, and Harris H Wang. Scalable and cost-effective ribonuclease-based rRNA depletion for transcriptomics. *Nucleic Acids Res.*, 48(4):e20, February 2020.
- Marcus B Jones, Sarah K Highlander, Ericka L Anderson, Weizhong Li, Mark Dayrit, Niels Klitgord, Martin M Fabani, Victor Seguritan, Jessica Green, David T Pride, Shibu Yooseph, William Biggs, Karen E Nelson, and J Craig Venter. Library preparation methodology can influence genomic and functional predictions in human microbiome research. *Proc. Natl. Acad. Sci. U. S. A.*, 112(45):14024–14029, November 2015.
- Martin Kircher, Susanna Sawyer, and Matthias Meyer. Double indexing overcomes inaccuracies in multiplex sequencing on the illumina platform. *Nucleic Acids Res.*, 40(1):e3, January 2012.
- Teemu Kivioja, Anna Vähärautio, Kasper Karlsson, Martin Bonke, Martin Enge, Sten Linnarsson, and Jussi Taipale. Counting absolute numbers of molecules using unique molecular identifiers. *Nat. Methods*, 9(1):72–74, November 2011.
- Amelie J Kraus, Benedikt G Brink, and T Nicolai Siegel. Efficient and specific oligo-based depletion of rRNA. *Sci. Rep.*, 9(1):12281, August 2019.
- Sung-Gwon Lee, Dokyun Na, and Chungoo Park. Comparability of reference-based and reference-free transcriptome analysis approaches at the gene expression level. *BMC Bioinformatics*, 22(Suppl 11):310, October 2021.
- Morten T Limborg, Antton Alberdi, Miyako Kodama, Michael Roggenbuck, Karsten Kristiansen, and M Thomas P Gilbert. Applied hologenomics: Feasibility and potential in aquaculture. *Trends Biotechnol.*, 36(3):252–264, March 2018.
- Alexandra A Mushegian, Roberto Arbore, Jean-Claude Walser, and Dieter Ebert. Environmental sources of bacteria and genetic variation in behavior influence Host-Associated microbiota. *Appl. Environ. Microbiol.*, 85(8), April 2019.
- Lasse Nyholm, Adam Koziol, Sofia Marcos, Amanda Bolt Botnen, Ostaizka Aizpurua, Shyam Gopalakrishnan, Morten T Limborg, M Thomas P Gilbert, and Antton Alberdi. Holo-Omics: Integrated Host-Microbiota multi-omics for basic and applied biological research. *iScience*, 23(8):101414, August 2020.

- Matthew R Olm, Christopher T Brown, Brandon Brooks, and Jillian F Banfield. drep: a tool for fast and accurate genomic comparisons that enables improved genome recovery from metagenomes through de-replication. *ISME J.*, 11(12): 2864–2868, December 2017.
- Askarbek Orakov, Anthony Fullam, Luis Pedro Coelho, Supriya Khedkar, Damian Szklarczyk, Daniel R Mende, Thomas S B Schmidt, and Peer Bork. GUNC: detection of chimerism and contamination in prokaryotic genomes. *Genome Biol.*, 22(1):178, June 2021.
- Donovan H Parks, Maria Chuvochina, Christian Rinke, Aaron J Mussig, Pierre-Alain Chaumeil, and Philip Hugenholtz. GTDB: an ongoing census of bacterial and archaeal diversity through a phylogenetically consistent, rank normalized and complete genome-based taxonomy. *Nucleic Acids Res.*, 50(D1): D785–D794, January 2022.
- Gianluca Prezza, Tobias Heckel, Sascha Dietrich, Christina Homberger, Alexander J Westermann, and Jörg Vogel. Improved bacterial RNA-seq by cas9-based depletion of ribosomal RNA reads. *RNA*, 26(8):1069–1078, August 2020.
- R Development Core Team. *R: a language and environment for statistical computing*. Vienna, Austria, 2008.
- Eugene Rosenberg and Ilana Zilber-Rosenberg. *The Hologenome Concept: Human, Animal and Plant Microbiota*. Springer, Cham, 2013.
- Andreas Schroeder, Odilo Mueller, Susanne Stocker, Ruediger Salowsky, Michael Leiber, Marcus Gassmann, Samar Lightfoot, Wolfram Menzel, Martin Granzow, and Thomas Ragg. The RIN: an RNA integrity number for assigning integrity values to RNA measurements. *BMC Mol. Biol.*, 7:3, January 2006.
- Michael Shaffer, Mikayla A Borton, Bridget B McGivern, Ahmed A Zayed, Sabina Leanti La Rosa, Lindsey M Solden, Pengfei Liu, Adrienne B Narrowe, Josué Rodríguez-Ramos, Benjamin Bolduc, M Consuelo Gazitúa, Rebecca A Daly, Garrett J Smith, Dean R Vik, Phil B Pope, Matthew B Sullivan, Simon Roux, and Kelly C Wrighton. DRAM for distilling microbial metabolism to automate the curation of microbiome function. *Nucleic Acids Res.*, 48(16): 8883–8900, September 2020.
- Christian M K Sieber, Alexander J Probst, Allison Sharrar, Brian C Thomas, Matthias Hess, Susannah G Tringe, and Jillian F Banfield. Recovery of genomes from metagenomes via a dereplication, aggregation and scoring strategy. *Nat Microbiol*, 3(7):836–843, July 2018.
- Yasuhiro Tanizawa, Takatomo Fujisawa, and Yasukazu Nakamura. DFAST: a flexible prokaryotic genome annotation pipeline for faster genome publication. *Bioinformatics*, 34(6):1037–1039, November 2017.

- Kevin R Theis, Nolwenn M Dheilly, Jonathan L Klassen, Robert M Brucker, John F Baines, Thomas C G Bosch, John F Cryan, Scott F Gilbert, Charles J Goodnight, Elisabeth A Lloyd, Jan Sapp, Philippe Vandenkoornhuyse, Ilana Zilber-Rosenberg, Eugene Rosenberg, and Seth R Bordenstein. Getting the hologenome concept right: an Eco-Evolutionary framework for hosts and their microbiomes. *mSystems*, 1(2), March 2016.
- Gherman V Uritskiy, Jocelyne DiRuggiero, and James Taylor. MetaWRAP—a flexible pipeline for genome-resolved metagenomic data analysis. *Microbiome*, 6(1):1–13, September 2018.
- Hsin-Jung Wu and Eric Wu. The role of gut microbiota in immune homeostasis and autoimmunity. *Gut Microbes*, 3(1):4–14, January 2012.
- Bo Zhang, Matthew Brock, Carlos Arana, Chaitanya Dende, Nicolai Stanislas van Oers, Lora V Hooper, and Prithvi Raj. Impact of Bead-Beating intensity on the genus- and Species-Level characterization of the gut microbiome using amplicon and complete 16S rRNA gene sequencing. *Front. Cell. Infect. Microbiol.*, 11:678522, October 2021.