Exploring the landscape of bioinformatic tools for fungal identification in shotgun metagenomic sequencing data

For potential applications in colorectal cancer biomarker discovery

Arfa Irej Qureshi

Master Thesis
Pharmacy
45 credits

Department of Pharmacy
The Faculty of Mathematics and Natural Sciences

UNIVERSITY OF OSLO

June / 2023

2023

Exploring the landscape of bioinformatic tools for fungal identification in shotgun metagenomic sequencing data: For potential applications in colorectal cancer biomarker discovery

Arfa Irej Qureshi

http://www.duo.uio.no/

Printing: Reprosentralen, Universitetet i Oslo

ABSTRACT

The collection of fungi inhabiting the human gut and their genes, known as the 'human gut mycobiome', have a significant impact on health and disease. While advancements in sequencing and computational technologies have expanded the study of microbial communities, existing tools primarily focus on bacterial data, with limited user-friendly options for analyzing fungal data in shotgun metagenomic datasets. Therefore, in order to meet the needs of a mycobiome characterization analysis, an optimized and automated pipeline is required.

This thesis aims to evaluate existing classification tools for fungal identification in shotgun metagenomic sequencing datasets for potential inclusion in a Snakemake pipeline that provides a taxonomic and functional classification of the mycobiome. To perform this evaluation, this thesis first presents five classification tools (Kraken 2, MetaPhlAn 3, FindFungi, HumanMycobiomeScan and FunOMIC) and their installation requirements. Two mock communities, the mixed mock community (bacterial, viral, fungal, and human genomes) and the fungal mock community (50 fungal genomes), are created to generate two simulated datasets, one for each mock community. The classification tools are then tested for their ability to classify fungal reads from the two simulated datasets.

The results show that Kraken 2 is able to classify fungal reads with varying levels of success based on the reference database used. When mapping against a custom reference database comprised exclusively of the 50 fungal genomes present in the fungal mock community, Kraken 2 is able to classify 99.83 % of the reads in the fungal mock community simulated dataset. MetaPhlAn 3 was able to classify three of the five (60 %) fungal species in the mixed mock community and 38 of 50 (76%) fungal species present in the fungal mock community. Technical challenges with the codes of FindFungi, HumanMycobiomeScan and FunOMIC hindered their implementation, thus evaluation of their fungal classification capabilities was not possible.

Based on the findings in this thesis, there is an urgent need to develop a robust pipeline to characterize the mycobiome accurately and efficiently in shotgun metagenomic datasets.

ACKNOWLEDGMENT

The completion of this thesis and the work presented within it would not have been possible without the help I received from a group of very important individuals. Their invaluable help and support have played a crucial role in guiding me throughout my final year at the University of Oslo.

Taking this into consideration, I would like to express my sincere appreciation to my supervisors Trine Ballestad Rounge and Hanne Cecilie Winther-Larsen. Throughout this final year, they have provided invaluable guidance, not only for this thesis but also for all the tasks I have undertaken. This past year would have been considerably more challenging without their support and commitment to helping me achieve my goals with this thesis.

I am also deeply appreciative of Ekaterina Avershina for her invaluable mentorship throughout the entire process. Her guidance, ideas, and innovative solutions gave me a solid foundation to complete this journey. She has assisted me in overcoming various challenges, particularly in the realm of bioinformatic analyses. Without her help, I would have likely found myself still grappling with command line operations and shell scripts. I am genuinely grateful for the time she has devoted to this project.

Furthermore, I am extremely thankful to Jenny Helene Mary Storvik Fjørtoft and Simon Nordvold Barak for the time they dedicated to helping me, all while successfully completing their own Master's theses. From jointly troubleshooting scripts, crafting RStudio scripts, creating Snakemake files, to offering moral support as I explored the world of bioinformatics, they have consistently assisted me with every difficult task I encountered.

Last, but by no means the least, I am truly grateful for the unwavering support of my amazing family. They have served as my guiding lights, inspiring and motivating me at every turn throughout my academic journey. Their enduring love, unwavering faith in my abilities, and constant encouragement have been pivotal in reaching this significant milestone. They have without fail provided the strength and reassurance I needed to overcome challenges and pursue my goals. As I venture into new horizons, I am confident that they will remain a pillar of support and strength for me.

TABLE OF CONTENTS

1 INTRODUCTION

1.1 BACKGROUND AND MOTIVATION

The human gut is home to trillions of microbes such as bacteria, archaea, viruses, phages, and fungi (1, 11, 12), collectively known as 'human gut microbiota' (1). The human gut plays a critical role in physiological functions and disease development (1). Emerging advancements in next-generation sequencing (NGS) and computational technologies have increased ways to study the compositional and functional characteristics of microbial communities associated with different body sites (2).

Many of the existing bioinformatics tools available today have been developed for the analysis of 16S ribosomal RNA (rRNA) bacterial data and are based on the amplicon sequencing method. Few tools employ shotgun metagenomic sequencing, a relatively new sequencing approach. Available tools for fungal classification in shotgun metagenomic datasets often lack user-friendliness and may not be optimally configured by default.

1.2 PROPOSED PIPELINE

An optimized and automated pipeline based on existing classification tools for fungal identification in shotgun metagenomic sequencing datasets is required. The pipeline should employ an extensive reference database to successfully map fungal reads with high specificity and sensitivity. The pipeline should also be user-friendly, modifiable and accommodate expansion so others may use it in their research and tailor it to their specific needs.

1.3 CLASSIFICATION TOOLS

To classify fungal reads in shotgun metagenomic datasets, sequencing reads have to be mapped against reference fungal genomes. Kraken 2 (3) and MetaPhlAn 3 (4) are often included in customized pipelines. One of the tools mentioned, FunOMIC (5), also performs functional annotation. Available classification tools for fungal identification in shotgun metagenomic datasets are listed in **Table 1**. This list is by no means exhaustive. These tools have been shortlisted for the purposes of this thesis.

Table 1: A selection of classifications tool used for fungal classification in shotgun metagenomic datasets.

Classification tool	Outcome of analysis
Kraken 2 (3)	Taxonomic profile
MetaPhlAn 3 (4)	Taxonomic profile
FunOMIC (5)	Functional annotation and taxonomic profile
FindFungi (6)	Taxonomic profile
HumanMycobiomeScan (7)	Taxonomic profile

2 THE HUMAN GUT MICROBIOME

The human gut microbiome (the collection of all human gut microbes and their genes) contains approximately 100 times more genes than the human genome (1). Additionally, the ratio of bacterial cells to human cells is close to 1:1 (1). Phages in turn far exceed the number of bacteria present. There are as many as 10-fold more phages than bacteria (8). These numbers illustrate the sheer magnitude of this community, and its subsequent potential to benefit and/or protect its host.

A distinction is made between the 'core human microbiome' and the 'variable human microbiome' (**Fig. 1**). An extensive gene catalog containing 9 879 896 genes showed country-specific differences in the microbial composition (9), suggesting environmental and lifestyle factors, such as diet, and even host genetics may affect the variable part of the microbiome. As part of the MetaHIT (Metagenomics of the Human Intestinal Tract) project, 124 fecal samples were collected from both healthy and obese individuals and inflammatory bowel disease (IBD) patients (10). Almost 35% of the genes in any one sample could be found in other samples – indicating the presence of a common 'core' genome (10).

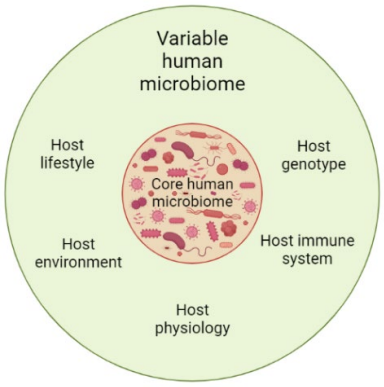


Fig. 1. *The core and variable human microbiome.* The core human microbiome represents the genes present at birth in all or the vast majority of humans across different populations, whereas the variable human microbiome is unique to different individuals based on various factors such as their environment, lifestyle, genetics and immune system (11). While the variable human microbiome is more transient in nature, the components of the core human microbiome are generally stable over time (12, 13). Created with BioRender.com.

Itai et. al (12) present two different approaches to describe the microbiome: the community-based and the function-based approach. The community-based approach focuses more on the species present, i.e., the taxa that are consistently present across the studied population (12) – here the human gut microbiome. A function-based definition is centered around the functional aspects of the genes present (12). Lozupone et. al (14) argue the latter may be a more realistic approach as it acknowledges that different species may play the same role in a niche.

As our knowledge of the composition of the microbiome and its function increases, so does our understanding of its role in human health. The human gut microbiome offers numerous benefits to its host, these include gut integrity (15), protection against pathogens (16), and regulation of host immunity (17). While a functioning microbiome is of great value to the host, a dysfunctional microbiome is equally devastating. Changes in the composition, and subsequent function, of the microbiome caused by genetic, dietary, and various environmental factors can change intestinal permeability and immune responses (18). Disturbances in the intestinal barrier can cause persistent activation of the immune system (15) and intestinal inflammation (19). Intestinal barrier dysfunction has also been implicated in a number of diseases such as celiac disease (20), IBD (15, 21) and colorectal cancer (CRC) (15).

2.1 FUNGI AS PART OF THE HUMAN GUT MICROBIOME

Fungi are a highly diverse microorganism in both their morphology and function. They belong to the kingdom eukaryotes (22). Despite being ubiquitous and an essential evolutionary contributor (23), their taxonomic classification remains largely unexplored. Of the total estimated 2.2-3.8 million fungal species that inhabit Earth, an insignificant portion (4%) has been cataloged (23). The lack of characterization may be due to several reasons, such as phenotypic diversity, genome plasticity, and the inability to culture most species (6).

The fungal component of the microbiome is called the 'mycobiome', a term first introduced by Ghannoum et. al in 2010 (24). Despite only constituting a small fraction, 0.1% of all microorganisms present in the gut, fungi play an important part in regulating human intestinal homeostasis and disease pathogenesis (25).

In terms of phyla, many studies (26-29) so far suggest that *Ascomycota* is the most predominant phylum found in the gut of healthy individuals, followed by *Zygomycota* and *Basidiomycota*. Some of the earliest work done on the composition of the mycobiome in healthy individuals found three fungal species *Galactomyces, Paecilomyces* and *Gloeotinia* to be persistent GI inhabitants (30). Scanlan and Marchesi thus concluded that the human gut mycobiome had low diversity and was relatively stable (30). While Hallen-Adams et. al also found low diversity and abundance in fungi (31), they argue that the mycobiome appeared unstable over time, in contrast to gut-associated bacteria (14). Suhr and Hallen-Adams postulate that there is little evidence to support the presence of a "core mycobiome" (28). According to Suhr and Hallen-Adams, an understanding of fungal physiology and ecology suggests even relatively common gut fungi, such as *Debaryomyces hansenii* and several

Penicillium species, are actually allochthonous, i.e., originating elsewhere, and are likely only passing through from dietary exposure without exerting influence on the gut mycobiome or host (28). To be considered a resident (autochthonous) of the human gut, they argue a fungus has to be able to grow at 37 °C (27). *Debaryomyces* and *Penicillium*, for example, are incapable of doing so (27). The contention on the topic matter amongst researchers underlines the importance of and the need for more studies that delve into this relationship.

2.2 ROLE OF HUMAN GUT MYCOBIOME IN HEALTH AND DISEASE

During the first year of life, gut fungal α-diversity, i.e., the number of species and their abundance within a community or the mean in a collection of communities (32), decreases while the bacterial reciprocally increases (33). One noteworthy finding of the study by Fujimura et. al (33) was that variations in the fungal β-diversity were a stronger predictor of the risk for atopy compared to changes in the bacterial community. This suggests that variations in the gut fungal community may be involved in affecting an infant's vulnerability to allergies and asthma during childhood. It is currently unclear whether the connection between the onset of immune diseases and the decline in microbial diversity is a result of a reduction in the number of microbial species or specifically due to the loss of crucial taxonomic or functional groups of microbes that are necessary for the proper development of the infant immune system (32).

Fungal dysbiosis, a term for altered composition of fungal communities (34), which includes a loss of symbionts, growth of pathobionts or opportunists and disturbed fungal diversity (25), has further been implicated in diseases in practically all parts of the body. These include autoimmune diseases such as irritable bowel syndrome (IBS) (35) and Crohn's disease (CD) (36), autism spectrum disorder (ASD) (37), obesity (29) and schizophrenia (38).

A study showed that children with type 1 diabetes (T1D) had a noticeably higher fungal species diversity despite no apparent difference in the total number of fungi present in both groups (39). Additionally, the incidence of *Candia albicans (C. albicans)* was lower in children with T1D than in healthy controls – 62% and 85% of all strains identified, respectively (39). On the other hand, the growth of *C. albicans* is seen in many diseases such as liver disease (40), asthma (41, 42), and COVID-19 (43). *C. albicans*, and other fungi, secrete prostaglandin-like oxylipin molecules (44), which are potent immunomodulatory molecules. This may provide a potential mechanism through which the growth of fungal species (e.g., *C. albicans*) in a mucosal site such as the gut may alter immune responses on the mucosa (42).

The number of opportunistic fungal infections in immunosuppressed individuals, e.g., HIV-positive individuals, those who have undergone organ transplantation or cancer chemotherapy, have significantly risen in the past two decades (45), indicating a susceptibility for opportunistic fungal infections amongst those with a compromised immune system.

3 COLORECTAL CANCER

3.1 EPIDEMIOLOGY

The amount of CRC cases globally are predicted to increase by 63 % by 2040 (46). CRC is the third most common cancer diagnosed and the second deadliest cancer with 935 000 deaths reported worldwide in 2020 (47). Locally, CRC is the third most common diagnosed cancer among both males and females in Norway (48). With an ever-rising number of cases being reported, there is an urgent need to figure out what predisposes people to CRC.

According to Global Cancer Observatory, CRC incidences are higher in the developed world (49). Epidemiological studies have shown strong associations with male sex (50). Besides sex, the risk of developing CRC increases with age (50), with 60.4% of all cases being diagnosed between the ages of 50 and 74 years (46). Genetics may also determine the risk of developing CRC. Approximately 5–10% of all CRC cases are hereditary CRC syndromes (51). CRC incidence and risk is not entirely dictated by demographic characteristics. Indeed, factors increasing CRC risk are modifiable, environmental lifestyle factors such as smoking (52), excessive alcohol intake (53), increased bodyweight (54), and red and processed meat (50, 55).

3.2 PATHOGENESIS

The discovery of various molecular pathways has revealed the heterogeneous nature of CRC. The classical pathway, henceforth referred to as the adenoma-carcinoma sequence, is the gradual advancement of healthy epithelial cells to abnormal, dysplastic cells and eventually to cancer, which results from the accumulation of several clonally selected genetic alterations (56, 57).

3.2.1 Genetic mutations, epigenetic alterations and aberrant signaling pathways

One of the hallmarks of cancer development is DNA damage. When faulty DNA repair leads to mutations and/or chromosomal abnormalities that negatively affect oncogenes and tumor suppressor genes, cells undergo transformation leading to malignant growth (58).

A significant type of genetic change found in colorectal tumors is the mutation of the *ras* gene. In the human body, three *ras* genes encode four closely related *ras* proteins. Mutated *ras* genes, when introduced into appropriate recipient cells, are capable of granting neoplastic characteristics (59). Of the three human *ras* isoforms, *Kirsten rat sarcoma* (*KRAS*),

is the most frequently altered gene with approximately 40% of all CRC cases harboring *KRAS* mutations (60, 61).

Epigenetics can broadly be described as heritable modifications in gene expression that are not driven by alterations in the DNA sequence. The typical epigenome of colon cancer contains hundreds to thousands of genes that have abnormal methylation, and it is believed that only a portion of these genes are responsible for the development and clinical characteristics of CRC (62). Scientists are currently studying the mechanisms that cause abnormal DNA hypermethylation and hypomethylation. One proposed explanation for increased methylation in tumor promoter regions is the upregulation of DNA methyltransferase (DNMT) expression (62).

Dysregulation of signaling pathways, such as the Wnt signaling pathway (regulates cell growth and differentiation of intestinal epithelial cells) and the epidermal growth factor receptor/mitogen-activated protein kinase (EGFR/MAPK) pathway (involved in cellular growth, proliferation, and survival of normal cells), can also contribute to the development of CRC (63, 64).

3.3 DIAGNOSIS AND TREATMENT

A CRC diagnosis is made based on factors such as clinical symptoms, laboratory tests, pathology, endoscopy, and imaging. Endoscopy is the method of choice for diagnosing CRC and can be conducted as a sigmoidoscopy or a total colonoscopy (65). Adenoma, non-cancerous tumors, detection rates are inversely associated with the risks of interval CRC (cancer developed after colonoscopy), advanced-stage interval cancer, and fatal interval cancer (66). This is presumed to be due to the detection of precancerous adenomas during endoscopy. In case of inadequate or incomplete endoscopy, computed tomography (CT) colonography is complementarily used for the diagnosis of polyps and CRC (50). It should be noted that CT imaging is not a standard procedure.

If a diagnosis is made, there are several options for the treatment and management of CRC. If the cancer is detected early, it may be possible to treat by resecting malignant polyps endoscopically all at once (50). Surgery remains the cornerstone of curative CRC treatment (50). A meta-analysis conducted to determine the oncological outcome of laparoscopic CRC surgeries found that unsuccessful laparoscopies seemed to be associated with adverse long-term perioperative outcome (67).

There are also many systemic treatments available, mainly for metastatic cancer where surgery is not an option. These are usually tailored to the individual with patient-

specific and disease-specific markers (50). Despite the development of modern medicine and numerous technological advancements, treatment of CRC continues to produce unsatisfactory outcomes, and as a result, mortality remains high. The method of CRC screening is therefore crucial to detect the cancer in a timely fashion as survival is best for non-metastasized disease, i.e., cancer that has not spread to other areas of the body (50).

3.4 EARLY DETECTION AND SCREENING

Early detection and screening are vital to prevent the development of colorectal cancer because the disease often does not present symptoms in its early stages. Colonoscopy is the most reliable way to prevent colorectal cancer (50). Though it is an invasive procedure, it has a high level of accuracy and provides the option of directly removing precancerous lesions and early cancer. People with elevated risk factors, such as those with a family history, chronic ulcerative colitis, prior adenomas, or colorectal cancer, are recommended to undergo regular colonoscopy for monitoring (50). Despite the limitations of current screening methods, randomized controlled trials have demonstrated that screening can decrease both the incidence (68-70) and mortality (68-71) of CRC.

3.4.1 The CRCbiome study

The CRCbiome study is a cohort study that examines the role of lifestyle and the gut microbiome in CRC screening participants (55). The primary aim of the study is to develop a classification algorithm that can detect advanced colorectal lesions by analyzing the gut metagenome, demographics, and lifestyle of the screened individuals.

Participants are recruited from the Bowel Cancer Screening in Norway (BCSN) trial (72). The BCSN study is designed as a randomized experiment that compares a single sigmoidoscopy screening with fecal immunochemical test (FIT) tests conducted every two years, with a maximum of four rounds (72). Those who received a positive FIT for occult blood were invited to participate in the CRCbiome study. Of the 2426 invited, 1413 (58%) agreed to participate (55). FIT measures for hemoglobin in the fecal sample (73), and a positive FIT sample is defined as hemoglobin >15 mcg/g feces (55). The reason for this inclusion criteria is that these participants are referred to a follow-up colonoscopy and will therefore have clinicopathological information available (55). The participants are invited to the CRCbiome study before their colonoscopy. They are then asked to give two more fecal samples 2- and 12-months post-colonoscopy (55).

DNA is extracted from the samples and sequenced into metagenomic datasets. These datasets are then analyzed using a customizable workflow manager (55). Publicly available tools, such as MetaPhlAn (74) and HUMAnN 3.0 (4), are utilized to carry out taxonomic classification and determine the microbial gene content, along with functional annotation (using gene ontology and Kyoto Encyclopedia of Genes and Genomes (KEGG) databases) (55). Abundance measures are used to compute taxonomic and functional alpha (within the sample) and beta (between samples) diversity, as well as serve as input for machine learning approaches that aim to create classifiers for high-risk individuals based on data analysis (55).

3.5 THE HUMAN GUT MYCOBIOME IN COLORECTAL CANCER

Fungal dysbiosis has been widely implicated in CRC and polyps showed an overall decreased fungal diversity compared to adjacent tissue (75-77). Coker et. al (75) discovered that the ratio of *Basidiomycota* to *Ascomycota* was greater in CRC patients compared to the control group. Another study had previously concluded with similar findings and noted an overall increase in the *Ascomycota* and *Basidiomycota* abundance in CRC patients (77). A study conducted amongst IBD patients found that this same ratio was also significantly different in patients with IBD than those in remission (78). Furthermore, it is believed *C. albicans* may potentially promote CRC development through the production of carcinogenic substances, inflammation, Th17 response and molecular mimicry (79). With chronic inflammation being a precursor for CRC (22, 50), these findings suggest that the *Basidiomycota* to *Ascomycota* ratio could serve as an indicator of fungal dysbiosis and provide a possible explanation for the role of fungal dysbiosis in the development of CRC.

Evidence points to several fungal species as the culprits for the disruption in the composition of the mycobiome. An increase in the opportunistic fungi *Trichosporon* and *Malassezia* was found to favor the progression of CRC (77). Preliminary results from a pilot study conducted in Kuala Lumpur, Malaysia identified the presence of a set of proteins secreted by *Schizosaccharomyces pombe* in stool samples from both CRC patients and healthy individuals (80). Results from the study show that the secretome proteins identified from the yeast were skewed towards the control samples compared with samples obtained from the CRC patients (80).

Though the role of fungi in the development of CRC, by promoting inflammation and disrupting the balance of the mycobiome, has recently gained traction, more research is needed to fully understand the relationship between the mycobiome and colorectal cancer.

4 SHOTGUN METAGENOMIC ANALYSIS – A RECIPE

There are primarily two culture-independent approaches used to characterize the mycobiome. The first approach, called amplicon sequencing, involves using a target, often the fungal rRNA gene locus (81), to show what species are present in a sample (**Fig. 2A**). This region includes the genes for the small-subunit (18S) and large-subunit (26S), an alternative to the bacterial 16S marker gene (82). These genes are divided by the internal transcribed spacer (ITS) regions ITS1 and ITS2 (82). rRNA subunits and ITS regions are used as phylogenetic markers to show what species are present in a sample (83). Although powerful, amplicon sequencing has its limitations. Firstly, due to various biases associated with PCR, it may miss a significant proportion of the diversity present in a community (84, 85). Secondly, amplicon sequencing provides insight only into the taxonomic composition of a microbial community, and it is not possible to directly determine the biological functions associated with these taxa using this method (86). Finally, amplicon sequencing can only be used to analyze taxa for which taxonomically informative genetic markers are known and can be amplified (86).

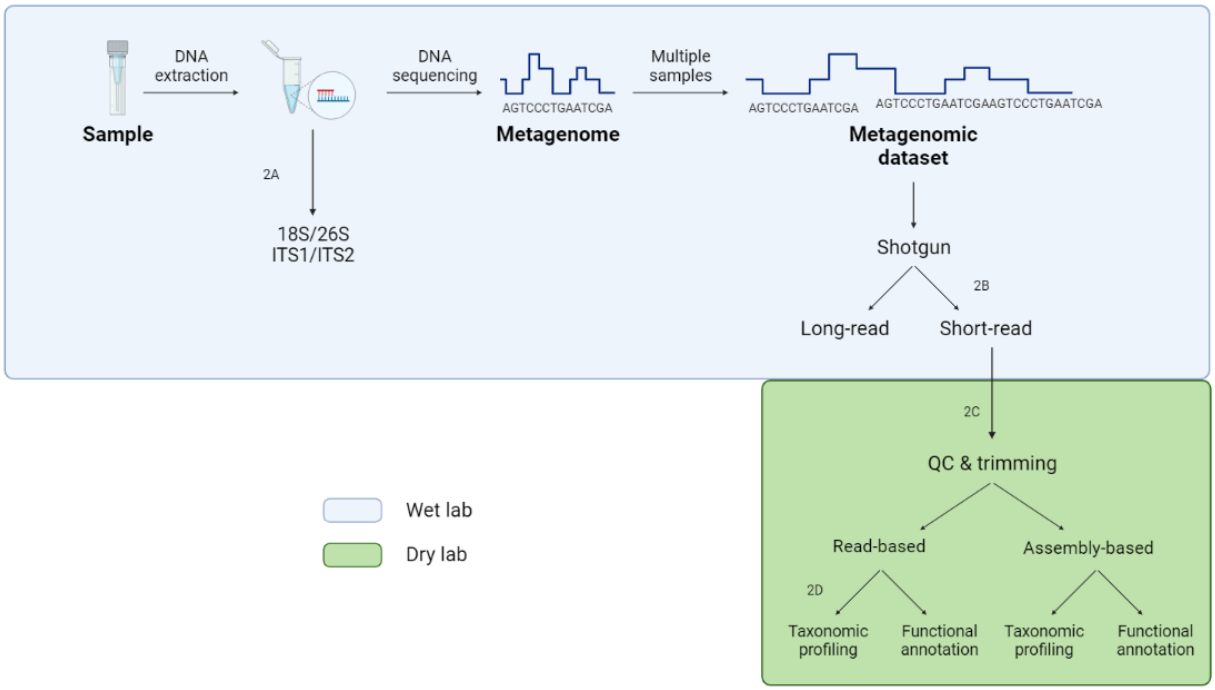


Fig. 2. *An overview of the steps involved in shotgun metagenomic analysis of fungi.* 2A: For amplicon sequencing, after the sample is collected, the extracted DNA from cells in the sample is amplified by polymerase chain reaction (PCR) and the 18S and/or 26S rRNA gene sequences, or the ITS regions ITS1 and ITS2, are used as phylogenetic markers. 2B: Short-read sequencing produces reads of shorter length, as the name suggests. Read lengths are usually up to 600 bases. 2C: Reads are typically pre-processed to remove low-quality bases, sequencing artifacts, and potential contaminations. 2D: Taxonomic sequence classifiers align or match input reads to nucleotide, protein, or whole genome databases to classify them. These classifications generate taxonomic annotations per read, which, when combined, form a taxonomic profile. Created with BioRender.com.

The second approach is called shotgun metagenomics. The term comes from the untargeted (*shotgun*) sequencing of all (*meta*) the genomes (*genomics*) present in the sample (87). Shotgun metagenomics entails subjecting DNA extracted from all the cells in a community to shotgun sequencing and circumvents a lot of the limitations of amplicon sequencing (83, 86). The large number of reads, i.e., base pairs sequenced from a DNA fragment, generated by shotgun sequencing allows for high resolution characterization of the microbial community, including the identification of rare taxa. Shotgun metagenomics can be used to examine the taxonomic makeup and potential functions of the microbial community (87), as it allows for the identification of genes and pathways present in the sample. It essentially reveals two main characteristics of the sample; what species are present in it and their function.

4.1 SAMPLE COLLECTION AND DNA EXTRACTION

The process of sample collection and DNA extraction detailed here, while also a part of amplicon sequencing, is in the context of shotgun metagenomics. DNA is extracted from cells with either mechanical lysis (bead beating) or chemical lysis which causes cell lysis, thus releasing the DNA (88). Bead beating is considered superior to chemical lysis (88). There are various kits for isolating DNA from a sample and they largely follow the same basic protocol with slight modifications based on the specimen used. Extraction of DNA in the CRCbiome study is carried out using the QIAsymphony automated extraction system, using the QIAsymphony DSP Virus/Pathogen Midikit (Qiagen), and each sample is lysed with bead beating (55). The efficiency of DNA extraction may vary depending on the type of sample and the microbial community being studied, leading to potential biases in the composition of the resulting metagenomic library. Wesolowska-Andersen et. al revealed a bias in the distribution of genes related to both taxonomy and function specific to the DNA extraction technique employed (89).

4.2 LIBRARY PREPARATION AND SEQUENCING

Lysis of the extracted DNA produces small DNA fragments. These DNA fragments are then independently sequenced producing *reads*. Reads can be *single-ended* or *pair-ended*. Paired-end reads are a pair of reads sequenced from the same DNA fragment in opposite directions. In contrast, in single-end reading, the sequencer reads a fragment from only one end to the other. While short-read sequencing results in shorter read lengths (up to 600 bases) (**Fig. 2B**),

long-read sequencing yields longer reads, usually more than 10 kilobases (90). Both approaches have their advantages and drawbacks, and the method of choice ultimately depends on the aim of the study. Short-read sequencing technologies are a popular choice due to their high throughput (91). There is also some evidence that suggests short paired-end reading is more cost effective than long-read sequencing (92).

The Illumina platform (Illumina HiSeq 2500 or 4000, NextSeq and NovaSeq) dominates shotgun metagenomics due to it being widely available, having remarkably high outputs and a high accuracy (87, 93). The Illumina platform is a short-read sequencing technology. The CRCbiome study employed Illumina NovaSeq for sequencing and read length for the paired-end run is 2 × 151 bp (55).

As the number of outputs achievable during a run is exceedingly high, several metagenomic samples can usually be sequenced at once by multiplexing of up to 96 or 384 samples (87). Multiplexing allows for multiple libraries to be pooled and sequenced together. A sample-specific index sequence (a "barcode") is added to DNA fragments during library preparation so reads can be identified and sorted before downstream analyses. This is typically done by using dual indexing barcode sets available for various library preparation protocols (87). The CRCbiome study prepared the sequencing libraries according to the Nextera DNA Flex Library Prep Reference Guide (v07) (Illumina, San Diego, CA, USA) (55).

4.3 QUALITY CONTROL AND TRIMMING

Before analysis, reads are typically filtered and trimmed (**Fig. 2C**) to remove low-quality reads and other artifacts that can affect downstream analysis. Reads are then generated as output data and usually formatted as a FASTQ file. FASTQ files are a standard format used to represent the output of high-throughput sequencing (HTS) technologies, such as Illumina, PacBio, and Oxford Nanopore, and their quality scores. These files contain the raw sequencing data in a four-line format for each read. The first line contains a sequence identifier, the second line contains the actual nucleotide sequence, the third line contains a separator, usually a plus sign (+), and the fourth line contains the corresponding quality scores for each base call in the sequence (94). The quality scores reflect the confidence of the base call at each position in the sequence and are represented using ASCII characters (94). The higher the quality score, the higher the confidence in the base call.

4.4 METAGENOME ASSEMBLY

Reads are used to construct a metagenome in metagenome assembly. Reads from the same genome are aligned and merged to create a longer string of reads called *contigs* (86, 95). Contigs are a continuous length of genomic sequence in which the order of the bases is known with a high degree of certainty. Contigs and *gaps* then together make up *scaffolds* (**Fig. 3**). Gaps arise when the sequences obtained from the two sequenced ends of a fragment overlap with reads from other fragments that are located in two separate contigs (96). The difference between contig and scaffold lengths thus correspond to the gaps within scaffolds.

Longer sequences provide information that is challenging to attain from analyzing unassembled raw reads. For example, the genome structure, the organization of genes into operons, and the regulatory promoters governing these. Nevertheless, the task of *de novo* metagenomic assembly is complicated and its success relies on factors such as the quantity of sequences and the diversity of the microbiome, including the abundance and uniformity of the present species (91). Though there is no consensus on how well the various assemblers perform in terms of important metrics such as completeness, continuity, and propensity to generate contigs (87), modern assemblers have somewhat alleviated this limitation (97).

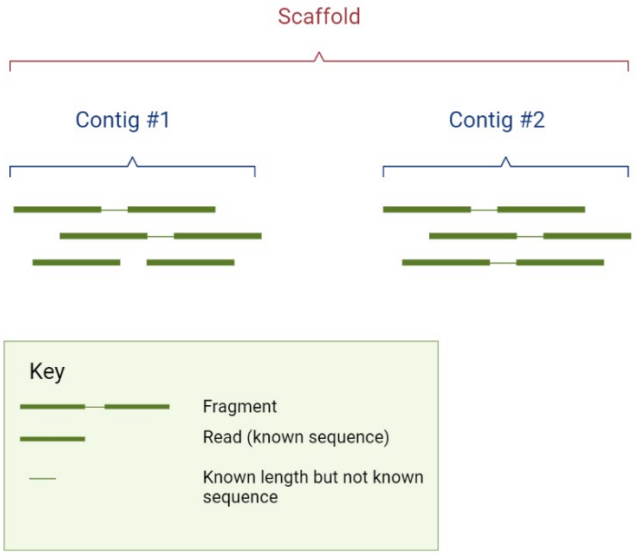


Fig. 3. *The hierarchy of scaffolds, contigs and reads.* Adapted from a figure found at https://mycocosm.jgi.doe.gov/help/scaffolds.jsf [Accessed 24.02.2023]. Created with BioRender.com.

4.5 READ-BASED PROFILING

Read-based profiling is a method where unassembled raw reads corresponding to genes in the reference database are used to determine the frequency of taxa and their functions, usually

employing a taxonomic classifier (**Fig. 2D**). The outcome of this extensive computational process is a collection of high-scoring pairs or matches that indicate potential similarities between genes in the dataset and genes in the reference database (98). An analysis is then conducted to derive a taxonomic profile and/or functional profile for the input data (98). Due to the aforementioned challenges with metagenome assembly, some authors opt to bypass the assembly step and instead move straight to the direct annotation of taxonomic and functional information from the raw reads (97).

4.6 LIMITATIONS OF SHOTGUN METAGENOMICS

Much like all sequencing technologies, shotgun metagenomics too comes with its own limitations. The analysis of shotgun sequencing data can be complex, requiring specialized bioinformatics expertise and access to high-performance computing resources. Though a rapid method to assess microbial populations, the information provided by shotgun metagenomics about the functional capabilities of these microorganisms is purely descriptive and lacks a prescriptive aspect (99). The presence of a gene may suggest the potential presence of a specific metabolic function, but it does not ensure that the associated microorganisms will actually exhibit the predicted biological activity. While shotgun metagenomics is an effective method for characterizing microbial communities, it is important to carefully consider the advantages and limitations of the technology in the context of the specific research questions and sample types being studied.

5 AIMS

5.1 PRIMARY AIMS

- Conduct a literature search for bioinformatic tools for fungal identification in shotgun metagenomic sequencing data.
- Validate the bioinformatic tools with a mock community.
- Shortlist the best methods for identifying fungal species in metagenomic sequencing data based on robustness and user-friendliness.
- Develop a Snakemake (100) workflow, consisting of the shortlisted bioinformatic tools, that identifies fungi in shotgun metagenomic sequencing data.

5.2 SECONDARY AIMS

- Taxonomically profile any fungal species present in the FIT samples collected from Norwegian CRC screening participants using the Snakemake (100) workflow.
- Functionally annotate any identified fungal species using the Snakemake (100) workflow.
- Study, identify and describe relationships between the human gut mycobiome and CRC based on the results from the functional annotation of the identified fungal species.

6 MATERIALS AND METHODS

All analyses, except for those performed in RStudio, were conducted on the supercomputer Saga. It is designed to run sequential and parallel workloads and became available for users in late 2019. RStudio was run on TSD. TSD is an offline high performance computing (HPC) cluster for handling sensitive data.

6.1 MOCK COMMUNITIES

During the evaluation process, two mock communities were used to generate simulated datasets to measure the performance of the classification tools for their ability to classify fungal reads in shotgun sequencing datasets.

6.1.1 Description

The first mock community (**Table 2**), hereafter referred to as the mixed mock community (MMC), contained five bacterial genomes, five fungal genomes, five viral genomes and the human genome. The MMC was created by one of the supervisors on the project.

Table 2: An overview of the species included in the mixed mock community.

Genome	Species	RefSeq ID
Fungi	*Klyuveromyces lactis*	GCF_000002515.2
	Saccharomyces cerevisiae	GCF_000146045.2
	Candida albicans	GCF_000182965.3
	Encephalitozoon intestinalis	GCF_000146465.1
	Malassezia restricta	GCF_003290485.1
Bacteria	*Escherichia coli*	GCF_000005845.2
	Clostridium perfringens	GCF_020138775.1
	Faecalibacterium prausnitzii	GCF_003312465.1
	Bacteroides fragilis	GCF_016889925.1
	Bifidobacterium longum	GCF_000196555.1
Virus	*Escherichia phage phiX174*	GCF_000819615.1
	Clostridium phage c-st	GCF_000865225.1
	Gokushovirinae Fen672_31	GCF_001190535.1
	Bacteroides phage B40-8	GCF_000883035.1
	Lambdavirus lambda	GCF_000840245.1
Human	*Homo sapiens*	GCF_000001405.40

The second mock community, henceforth referred to as the fungal mock community (FMC), consisted of fifty fungal genomes. FMC was constructed by downloading fungal

genomes from https://www.ncbi.nlm.nih.gov/genome/ on February 7th, 2023. The sequences were downloaded from RefSeq in FASTA format (.fna). In total, 50 fungal genomes were collected (1.3 GB). The accession numbers of the fungal genomes are listed in **Appendix I**.

6.1.2 Generation of simulated datasets

Two Illumina (HiSeq 2500) simulated datasets, one for each mock community, were generated using ART (101) latest version ART-MountRainier-2016-06-05.

The MMC simulated dataset was generated by one of the supervisors on the project. To generate a simulated dataset from the FMC, the fifty FASTA files for the fungal genomes were concatenated with the Python script 'prepare_fasta.py' (**Appendix II**). The Python script was curated in-house by the same supervisor who generated the MMC and the MMC dataset. The resulting concatenated FASTA file was then used to generate the FMC simulated dataset by executing the following command in ART (101):

```
$ ./art_illumina -ss HS25 -i ~/filename.fna -l 150 -p -m 500 -s 50
-c 1000000 -o filename
```

The parameters of the command are described in **Table 3**. The outputs are reads in FASTQ format and ALN (alignment) format. As this is a paired-read simulation, two files are produced for each format, giving a total of four files. The files, mock_fungal_paired1.fq/aln and mock_fungal_paired2.fq/aln, contain data of the first reads and for the second reads, respectively.

Table 3: Parameters chosen for the fungal mock community (FMC) simulated dataset.

Parameter	Description	Value
-ss --seqSys	Name of the Illumina sequencing system	HiSeq 2500 (125bp, 150bp)
-i --in	Filename of input DNA reference	~/mock_fungal.fna
-l --length	Length of reads	150 basepairs (bp)
-p --paired	Indicates a paired endread simulation	-
-m --mflen	Mean size of DNA fragments for paired end read simulations	500
-s --sdev	Standard deviation of the DNA fragments for paire end read simulations	50
-c --rcount	Number of reads to be generated per sequence	1 000 000
-o --out	Prefix of output file	mock_fungal_paired

6.2 METAPHLAN 3

MetaPhlAn 3 (Metagenomic Phylogenetic Analysis) (4) is a bioinformatic tool that identifies the microbial composition of a sample based on marker genes. It estimates the relative abundance of species by mapping reads against a collection of clade-specific marker sequences. These sequences are selected from coding sequences that identify microbial clades at the species level or at higher taxonomic levels. MetaPhlAn 3 (4) maps reads from a given sample to a catalog spanning over 1 million markers for 13 475 species using bowtie2 (102). Reads that belong to clades without an available genome are marked as an 'unclassified' subclade of their closest ancestor (for which data is available) (74). Clade abundances are estimated by normalizing read-based counts by the average genome size of each clade (74). It has a classification rate of about 10 000 reads per second (4), thus providing robust high-throughput assessments of metagenomic data at the species level. MetaPhlAn 3 (4) uses 2.6 GB memory for a complete taxonomic profiling run (4).

6.2.1 Mapping the simulated datasets

MetaPhlAn 3 (version 3.0.7) (4) was loaded into the environment with the following command:

```
$module load MetaPhlAn/3.0.7-foss-2020b
```

To map the raw reads of the simulated dataset from the MMC in MetaPhlAn 3 (4), the following commands were executed:

```
$metaphlan -input_type fastq ~/inputfile1.fq -o filename1.txt
```

```
$metaphlan -input_type fastq ~/inputfile2.fq -o filename2.txt
```

The first argument -input_type is used to specify the type of input while the second argument is the path to the input file. Lastly, -o is used for the name of the output file. Similarly, the following commands were executed to map the raw reads of the FMC simulated dataset:

```
$metaphlan -input_type fastq ~/inputfile1.fq -o filename1.txt
$metaphlan -input_type fastq ~/inputfile2.fq -o filename2.txt
```

6.2.2 Visualizing the MetaPhlAn 3 output

The phyloseq (103) package in RStudio was used to visualize the output from MetaPhlan 3 (4). The two MetaPhlan 3 (4) outputs from the MMC simulated dataset were merged using the script 'merge_metaphlan_output.R' (**Appendix III**). The merged output was then converted to a bar plot in phyloseq using the script 'Metaphlan2Phyloseq.R' (**Appendix IV**). The same process was repeated for the FMC simulated dataset.

6.3 KRAKEN 2

Kraken 2 (3) is a fast and accurate tool that uses *k*-mer-based algorithms to classify reads to the lowest possible taxonomic rank. *K*-mers are short genomic substrings of length *k* and are made up of nucleotides, i.e., A, T, C and G. They are used to identify species in metagenomic samples. Kraken 2 (3) was developed as an improvement of the large storage requirements of Kraken (104). Kraken's (104) default database can easily exceed 100 GB, especially when eukaryotic genomes are included in the reference database. Storage usage has been reduced by about 85% for Kraken 2 (3) while maintaining the same accuracy. Additionally, Kraken 2 (3) processes reads at higher speeds than Kraken (104) at 93.2 Mreads/min compared to 18.4 Mreads/min, respectively. Kraken 2 (3) can be installed by downloading its source code and manually installing the program or through a Conda environment. The source code is open-source and available in a GitHub repository at https://github.com/DerrickWood/kraken2.

6.3.1 Curating custom databases

The standard Kraken 2 (3) database, containing archaea, bacteria, viral, plasmid, human, UniVec_Core, does not include fungal genomes and thus does not fulfill the needs of fungal reads analyses. A custom Kraken 2 database called PlusPF was downloaded from https://benlangmead.github.io/aws-indexes/k2. The database contains the standard Kraken 2 (3) database plus protozoa and fungi. It has an index size of 69 GB. The contents of the

.tar.gz file were extracted and placed into a specific directory by executing the following command:

```
$ tar -xvzf filename.tar.gz -C ~/kraken2_analysis
```

In addition to downloading the PlusPF Kraken 2 (3) database, an exclusively fungal custom database was also curated. This database was called 'FungiDB'. The first step in building a custom Kraken 2 (3) database is installing a taxonomy. This was done with the command:

```
$ kraken2-build --download-taxonomy --db FungiDB
```

The parameter '--db' denotes the database name. The fungi library provided by Kraken 2 (3) developers was then installed with the command:

```
$ kraken2-build --download-library fungi --db FungiDB
```

Finally, to build the database, the following command was executed:

```
$ kraken2-build --build --db FungiDB --threads 24
```

No custom *k*-mer length (*k*) or minimizer length (*l*) were specified. The default values of *k* and *l* are 35 and 31, respectively.

6.3.2 Using a custom database as a positive control

To test if Kraken 2 (3) can accurately classify all species present in the FMC, a custom database called 'FMC_DB' was curated. This database contained all fifty fungal species included in the FMC. The analysis run with FMC_DB as the reference database served as a positive control against the other Kraken 2 (3) runs.

6.3.3 Mapping the mixed mock community

The following command was used to classify the raw reads from the MMC simulated dataset against the PlusPF Kraken 2 (3) database:

```
$ kraken2 --db ~/PlusPF/ --threads 16
--unclassified-out filename#.fq
--classified-out filename #.fq
--output filename.txt
--report filename.txt
--paired --use-names ~/art_dataset_100000_even1.fq
~/art_dataset_100000_even2.fq
```

The options of the command are described in **Table 4**.

Table 4: The options used to classify sequences from the mixed mock community (MMC) simulated dataset in Kraken 2 (3).

Option	Description
--db	Name of the Kraken 2 (3) database
--threads	Number of threads
--unclassified-out	Send unclassified reads to a file
--classified-out	Send classified reads to a file
--output	Output file
--report	Sample report file
--paired	Indicates that the input files are paired read data
--use-names	Replaces the taxonomy ID column with the scientific name and the taxonomy ID in parenthesis

It should be noted that paired read data requires the addition of "#" in the filenames in the --unclassified-out and --classified-out options. Kraken 2 (3) will replace this with "_1" and "_2" and spread the paired reads across the two files.

6.3.4 Classifying raw reads from the fungal mock community dataset

The FMC simulated dataset was mapped against all three custom databases by executing the following command:

```
$ kraken2 --db $database
--threads 16 --unclassified-out filename#.fq
--classified-out filename#.fq
--output filename.txt
--report filename.txt
--paired --use-names ~/mock_fungal_paired1.fq ~/mock_fungal_paired2.fq
```

The argument $database is the database of interest (PlusPF/FungiDB/FMC_DB).

6.3.5 Generating Krona charts

Krona (105) charts are a type of pie chart with multiple layers that are commonly employed in metagenomic visualization for examining data in a phylogenetic hierarchy. By using the output from Kraken 2 (3), these charts can be created to represent the proportion of reads originating from various taxonomic ranks as percentages. Krona (105) charts can be viewed with any web browser and thus does not require installation. The Python script 'kreport2krona.py' converts a Kraken 2 (3) report to a Krona (105) compatible .txt file. The

following command was used to convert the Kraken 2 (3) report for the MMC simulated dataset to a Krona (105) compatible .txt file:

```
$ ./kreport2krona.py -r ~/filename.txt -o filename.txt
```

The parameter -r is the Kraken 2 (3) report and -o denotes the output file. The output .txt file can be imported into Krona using the option ktImportText with the following command:

```
$ ktImportText filename.txt -o filename.html
```

The first argument is the path to the Krona-compatible .txt file and the second argument is the output file, i.e., the Krona (105) chart, in HTML format.

6.4 FINDFUNGI

FindFungi (6) is a pipeline that identifies fungal sequences in metagenomic datasets and taxonomically classifies them. Donovan et. al compared five algorithms (BLAST (106), DIAMOND (107), Kaiju (108) and two versions Kraken (104)) to identify the best method for identifying fungal species. They constructed a test database with nine bacterial genomes and one fungal genome. These were then used to generate three simulated metagenomic datasets with ART (101) to test the aforementioned algorithms. Their findings demonstrated that Kraken showed the highest sensitivity with all three datasets, albeit with lower specificity than the three other methods.

To identify fungi in metagenomic datasets, Donovan et. al applied FindFungi (6) to a total of 70 datasets. The FindFungi (6) pipeline identified 77 fungal species in 39 of these datasets. To determine whether these included any false positives, Donovan et. al compared the results to NCBI nt/nr database by using BLAST (106). Additionally, to minimize these false positives, there is a read distribution step in the FindFungi (6) pipeline. A fungal genome reference database was constructed by downloading all fungal genomes from GenBank.

The FindFungi (6) pipeline analyzes raw sequences in a FASTQ format, then uses Skewer (109) to remove low quality reads, and finally the remaining reads are converted into FASTA format. Reads predicted as non-fungal are removed. The best hit for each of the reads is then mapped to a pseudo-assembly of the relevant genome using BLAST (106).

6.4.1 Installation

All scripts were downloaded from https://github.com/GiantSpaceRobot/FindFungi and added to the ~/findfungi_analysis directory. The FindFungi (6) pipeline requires the installation of the following external programs to be able to run:

- gcc version 4.4.4 20100726 (Red Hat 4.4.4-13) (Tested with version 10.3.0)
- coreutils 8.27 (Tested with version 8.32)
- python 2.7.13 (Tested with version 3.9.5)
- skewer 0.2.2
- kraken 0.10.5-beta (Tested with version 1.1.1)
- ncbi blast 2.2.30 (Tested with BLAST+ version 2.2.31)
- Rscript 3.3.3 (packages: wordcloud) (Tested with version 4.1.0)
- graphviz 2.40.1 (Tested with version 2.47.2)

All the dependencies were loaded into the environment using the command:

$module load [name of the external program]

6.4.2 Implementation

A SLURM (Simple Linux Utility Resource Management) script was used to run the pipeline. SLURM is an open-source job scheduler for Linux and Unix-like operating systems. The pipeline was tested with the MMC simulated dataset as the input file. The job was submitted to SLURM with the command:

$ sbatch ~/filename.sh

The following command was given within the SLURM script to execute the pipeline:

$ ~/FindFungi-0.23.3.sh ~/inputfile.fq filename

The first argument provided in the command line, after adding the path to the FindFungi (6) script, indicates the location of the input FASTQ file. The second one involves assigning a descriptive name to this dataset that FindFungi (6) will utilize.

6.5 HUMANMYCOBIOMESCAN

HumanMycobiomeScan (7) is a tool that classifies metagenomic sequencing reads and assigns them to specific fungal taxa by using a reference database of fungal genomes. The reference database is based on the complete fungal genomes available at NCBI website (7). The reference database was constructed in February 2018 (7), and may not have been updated since. HumanMycobiomeScan (7) is specifically designed to detect fungi that are commonly found in the human body. Still, the databases can be customized to the user's needs, thus making the program capable of working with datasets of various origins (e.g., fungal genomes associated with soil, water, air etc.) (7).

The workflow of HumanMycobiomeScan (7) begins with metagenomic reads being aligned to the fungal genome database using bowtie2 (102) to reduce the sample size by

removing sequences that do not match the reference database. Reads shorter than 60 bp are discarded (7). A double-filtering step is also included in the workflow to remove any possible contamination by human and bacterial sequences, as the inputs may stem from human-associated samples such as feces or tissues. Finally, filtered reads are matched again to the fungal database using bowtie2 (102) for taxonomic assignment. An additional step allows users to normalize results by the length of the references in the database (7). HumanMycobiomeScan (7) is available for download on the website: http://sourceforge.net/projects/hmscan.

6.5.1 Installation

The HumanMycobiomeScan tool was downloaded from https://sourceforge.net/projects/hmscan/ and added to the directory ~/hms_analysis. The tool requires a number of programs to run:

- bowtie2 (Tested with version 2.4.2)
- Samtools (Tested with version 1.11)
- R (Tested with version 4.0.3)
- BLAST+ (Tested with 2.11.0)

The programs were all tested with the latest version available in Saga at the time of the analysis and were loaded into the environment using the command:

```
$module load [name of the external program]
```

6.5.2 Creating a custom database

The developers suggest creating a custom database despite a small database being included in the tool to cover the needs of the user's analyses. A total of 2907 fungal genomes were downloaded from https://www.ncsbi.nlm.nih.gov/genome/browse#!/eukaryotes/fungi in a .csv file. The 'Assembly' column of this file was copied to a .txt file and uploaded to https://www.ncbi.nlm.nih.gov/sites/batchentrez. The 'Assembly' database with GenBank source was selected and a compressed FASTA archive for the database was downloaded. The size of this database was 26.3 GB. The database was created with the provided 'custom_database_creation_large.sh' script and submitted to SLURM with the command:

```
$sbatch hmscustomdb.sh
```

The SLURM script 'hmscustomdb.sh' can be found in **Appendix V**.

6.5.3 Implementation

Within a SLURM script, the following command was given to run the tool:

```
$ ./MScan.sh -p 16 -m ~/HMS/ -d ~/database
-1 ~/inputfile1.fq -2 ~/inputfile2.fq -o ~/filename
```

The arguments provided in the command are explained in **Table 5**.

Table 5: The parameters for the implementation of HumanMycobiomeScan (7).

Parameter	Description
-p	Number of threads
-m	Path to HMS directory
-d	Path to database
-1	Path to input file (if paired end)
-2	Path to input file (if paired end)
-o	Output folder

6.6 FUNOMIC

FunOMIC (5) is a pipeline that maps shotgun sequencing reads to reference databases to obtain the taxonomical profile of the mycobiome and functionally annotate it. The pipeline includes two built-in fungal databases FunOMIC-T and FunOMIC-P for taxonomical and functional annotation, respectively. FunOMIC-T contains 1.6 single-copy genes from almost 5000 fungal genomes, while FunOMIC-P contains more than 3 million fungal proteins. FunOMIC-T is comprised of eight phyla. Of these, *Ascomycota, Basidiomycota* and *Mucoromycota* represent more than 98% of the genomes (5). Its source code is freely available for download at https://github.com/ManichanhLab/FunOMIC. It is worth noting that FunOMIC (5) is the first tool of its kind to also provide functional annotation in addition to taxonomic profiling.

6.6.1 Installation

To successfully implement the FunOMIC (5) pipeline, a number of dependencies need to be installed:

- bowtie2 (Tested with version 2.4.4)
- Samtools (Tested with version 1.13)
- FLASH2 (Tested with version 2.2.0)
- DIAMOND (Tested with version 2.0.15)
- R – KEGGREST package (Tested with version 4.1.0)

All the dependencies were loaded into the environment using the command:

```
$module load [name of the external program]
```

The pipeline was downloaded from https://github.com/ManichanhLab/FunOMIC and added to the directory ~/funomic_analysis.

6.6.2 Implementation

The pipeline was tested by executing the following command:

```
$./FunOMIC.sh -1 ~/mock_fungal_paired1.fq -2 ~/mock_fungal_paired2.fq
-p mock_fungal -o mock_fungal_output -a ~/BacterialDB/ -b ~/FunOMIC-Tv1/
-c ~/FunOMIC.P.v1/ -t 16
```

The arguments provided in the command are explained in **Table 6**.

Table 6: The parameters for the FunOMIC (5) pipeline execution.

Parameter	Description
-1	Path to input file
-2	Path to input file
-p	Output prefix
-o	Output folder
-a	Path to bacterial database
-b	Path to FunOMIC-T database
-c	Path to FunOMIC-P database
-t	Number of threads

The FunOMIC-T, FunOMIC-P and the bacterial databases were downloaded from https://manichanh.vhir.org/funomic/.

6.7 SNAKEMAKE

Workflow managers in shotgun metagenomic analysis are software tools that help automate and streamline the process of analyzing large volumes of genomic data generated from metagenomic sequencing experiments. These workflow managers help in managing the complexity of the analysis pipeline and facilitate the efficient execution of various analysis steps. The tools allow users to configure and customize the analysis steps as per their requirements.

Snakemake (100) is a workflow management system that enables users to execute all the steps involved in data analysis, starting from raw data processing to plotting results in graphs and tables. Snakemake's (100) main concept is that workflows are defined by breaking

them down into steps that are depicted as *rules*. Each rule outlines the process of obtaining a collection of output files from a collection of input files (100). Rules are transformed into a job that generates the specified output files. Job dependencies are not explicitly defined but are inferred automatically. Snakemake (100) identifies a rule that can generate each input file for a job, creating another job accordingly. This process is repeated recursively until all input files for all jobs are either generated by another job or already exist in storage. Snakemake (100) then uses this inference to automatically generate a directed acyclic graph (DAG) that represents the dependencies between the different analysis steps.

Snakemake (100) uses a Python-based domain specific language (DSL). It also supports the integration of external tools and scripts, which allows users to incorporate existing analysis pipelines into a Snakemake (100) workflow. When running a data analysis workflow, the runtime and resources utilized are primarily influenced by the jobs that are executed and the effectiveness of the libraries and tools utilized in these jobs. Snakemake (100) was primarily chosen for this project due to the available resources in the research group. Though it is entirely possible similar analyses can be conducted with Nextflow (110).

7 RESULTS

To evaluate how well a selection of classification tools were able to classify fungal reads, two simulated datasets were generated. The five classification tools (MetaPhlAn 3 (4), Kraken 2 (3), FindFungi (6), HumanMycobiomeScan (7) and FunOMIC (5)) were tested on each of these datasets.

7.1 METAPHLAN 3

MetaPhlAn 3 (4) was unable to classify all the fungal reads from the two simulated datasets. Of the five fungal species present in the MMC simulated dataset, only three of these (*C. albicans, Malassezia restricta* and *S. cerevisiae*) were classified by MetaPhlAn 3 (4) with the provided MetaPhlAn 3 (4) database. A higher proportion of fungal reads in the FMC simulated dataset were classified. Of the fifty fungal species in the FMC simulated dataset, MetaPhlAn 3 (4) was able to classify thirty eight of these. This gives a total of 76 % classified species in the FMC simulated dataset compared to 60% of classified species in the MMC simulated dataset. MetaPhlAn 3's (4) classification of species from the MMC and FMC simulated datasets is presented in **Fig. 4A** and **Fig. 4B**, respectively.

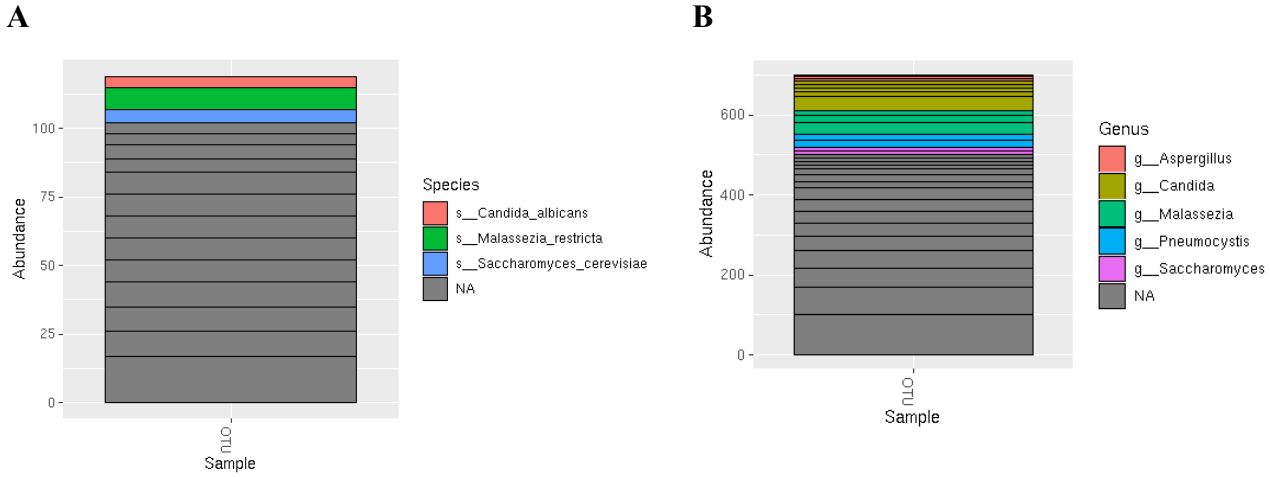


Fig. 4. *Classification of the two simulated datasets by MetaPhlAn 3 (4).* 4A: MetaPhlAn 3 (4) classified three out of five fungal species from the mixed mock community (MMC) simulated dataset. 4B: The genera MetaPhlAn 3 (4) was able to classify from the fungal mock community (FMC). Results are visualized in RStudio with the phyloseq (103) package.

7.1.1 Relative abundance

The relative abundance of the kingdom Eukaryota in the MMC simulated dataset was 17.38%. The relative abundances of the three classified fungal species in the MMC simulated dataset are presented in **Table 7**.

Table 7: The relative abundances of *Candida albicans, Malassezia restricta* and *Saccharomyces cerevisiae* in the mixed mock community (MMC) simulated dataset.

Species	**Relative abundance**	
	Theoretical	MetaPhlAn 3 (4)
Candida albicans	6.25	4.41398
Malassezia restricta	6.25	8.1684
Saccharomyces cerevisiae	6.25	4.79846

In the FMC simulated dataset the relative abundance of the kingdom Eukaryota was 100%. The relative abundances of the five classified fungal genera in the FMC simulated dataset are presented in **Table 8**.

Table 8: The relative abundances of the fungal genera *Aspergillus, Candida, Malassezia, Pneumocystis* and *Saccharomyces* in the fungal mock community (FMC) simulated dataset.

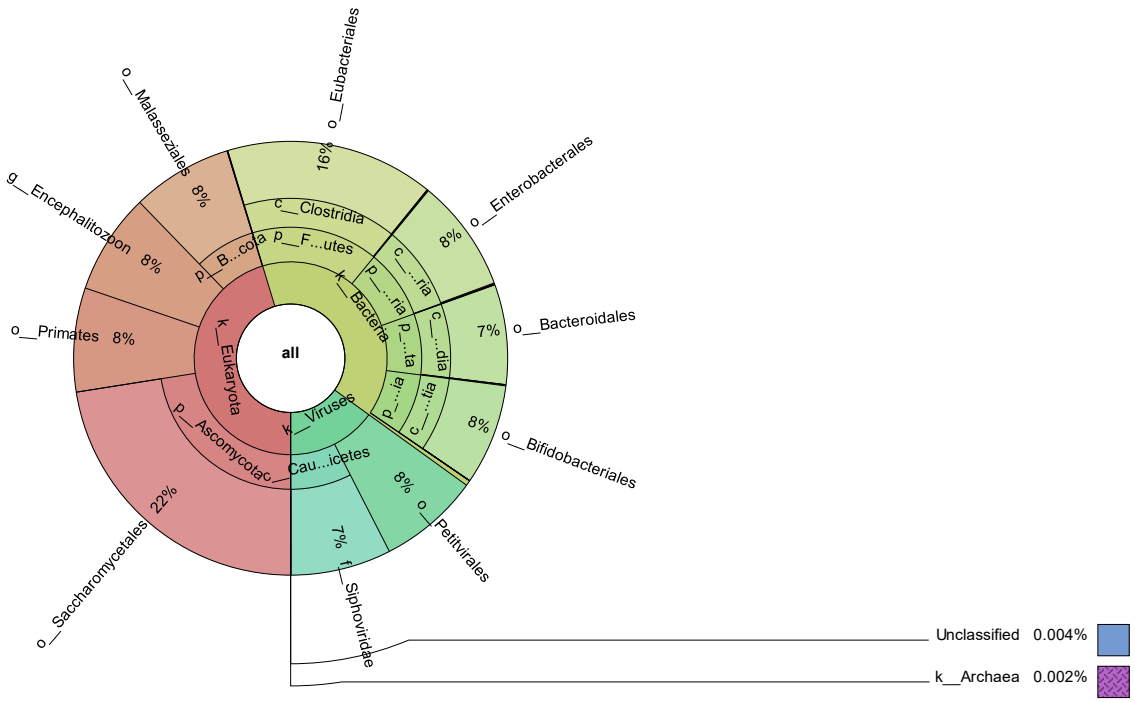
Genus	**Relative abundance**	
	Theoretical	MetaPhlAn 3 (4)
Aspergillus	4	7.92668
Candida	14	35.766155
Malassezia	4	29.87563
Pneumocystis	2	16.31203
Saccharomyces	2	10.119515

7.2 KRAKEN 2

7.2.1 Classification of the mixed mock community

Mapping the MMC simulated dataset against the PlusPF custom Kraken 2 (3) database produced 99.99% classified reads (**Fig. 5A**). The fungal fraction, presented in **Fig. 5B**, showed proportionate levels of classification at 17%, suggesting that almost all of the reads were successfully classified by Kraken 2 (3).

A



B

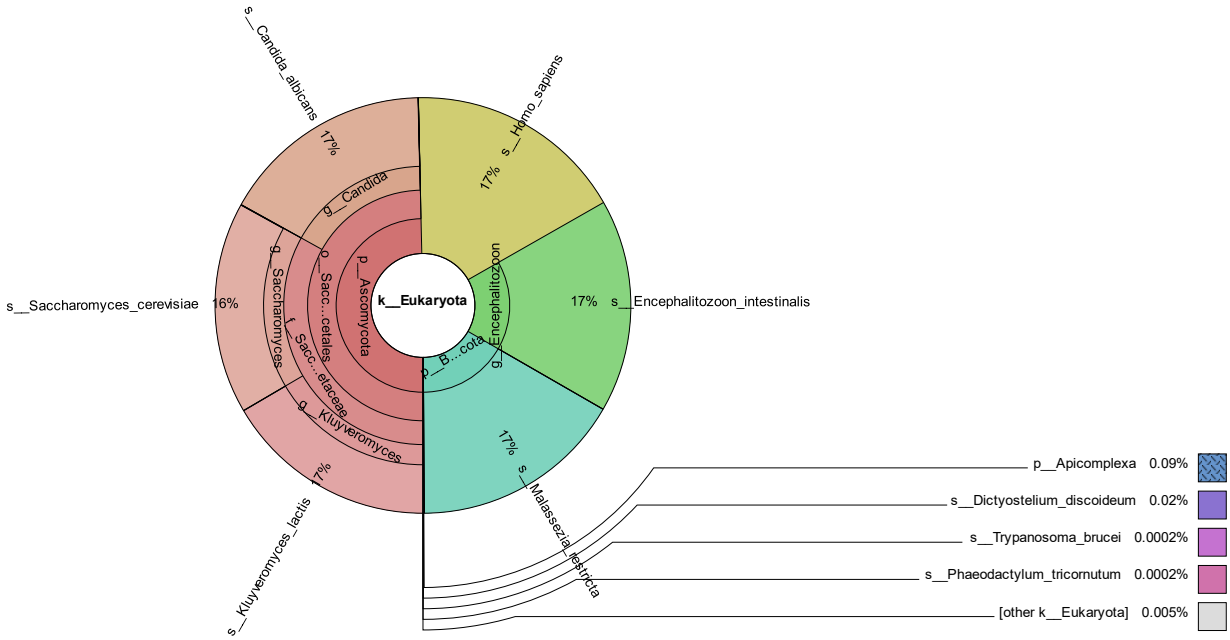


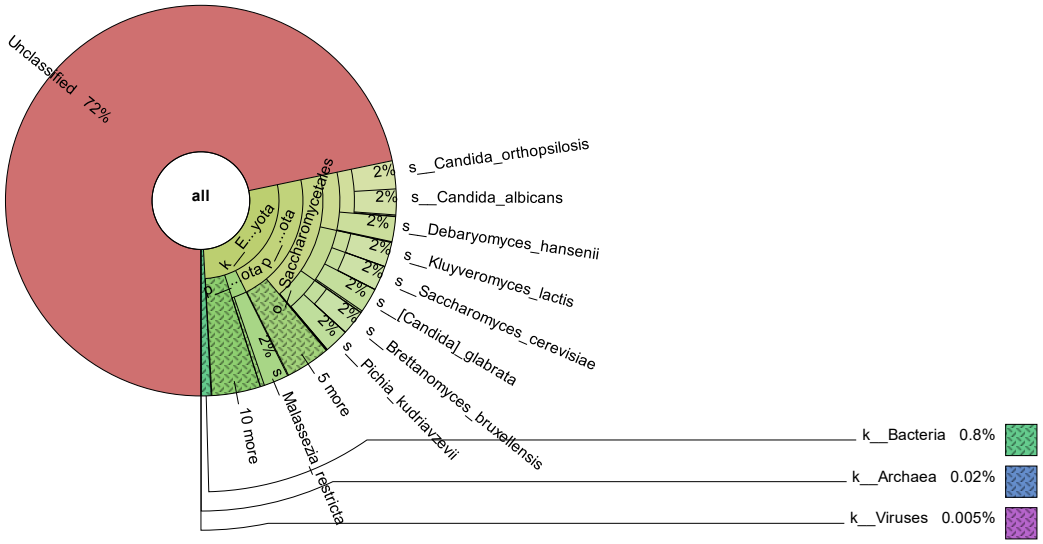
Fig. 5. *Classified species from the mixed mock community (MMC) simulated dataset.* 5A: Kraken 2 (3) was able to classify all the bacterial and fungal species present in the MMC. Only two of the five viral species were classified. 5B: The classified fungal fraction of the MMC. The reference database used for this classification was a custom database (69 GB) curated by Kraken 2 (3) developers, and contained fungi and protozoa in addition to the standard Kraken 2 (3) database (archaea, bacteria, viral, plasmid, human, UniVec_Core). Results were visualized using KronaTools (105).

7.2.2 Classification of the fungal mock community

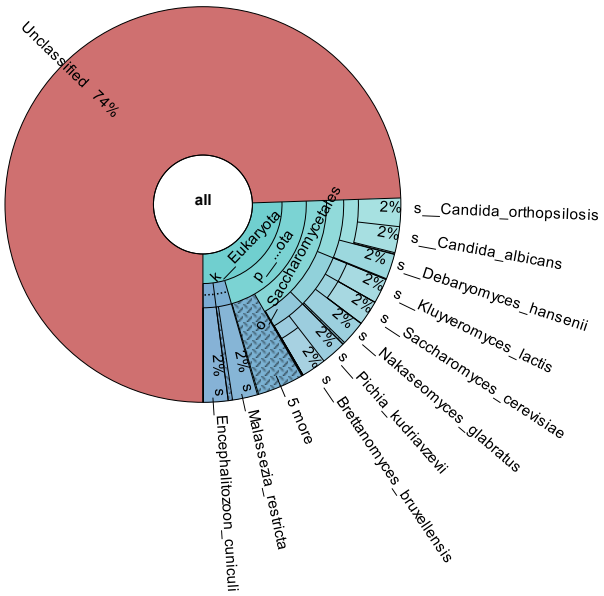
A large majority of the FMC simulated dataset (72%) remained unclassified when mapped against the PlusPF Kraken 2 (3) database (**Fig. 6A**). When using a custom reference database

comprising exclusively of fungal genomes, Kraken 2 (3) classified about 26% of the reads from the FMC simulated dataset (**Fig. 6B**). This contradicts the initial expectation of higher classification rates since the reference database is exclusively fungal. The results also suggest that this reference database may lack some of the fungal genomes included in the PlusPF database. Though when the reference database was customized to include all the fifty fungal species present in the FMC simulated dataset, Kraken 2 (3) was able to classify 99.83% of the reads from the FMC simulated dataset (**Fig. 6C**).

A



B



C

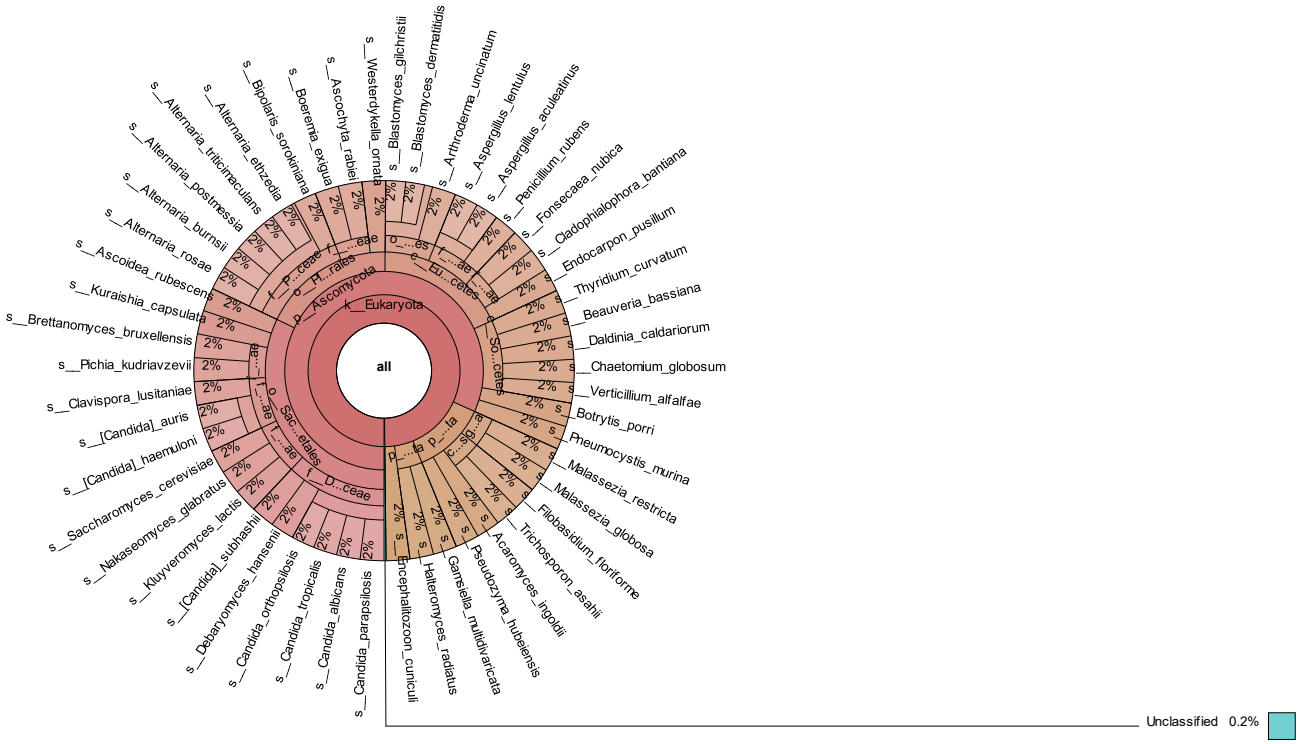


Fig. 6. *Classified species from the fungal mock community (FMC) simulated dataset.* 6A: The reference database used for this classification was a custom database called PlusPF (69 GB) curated by Kraken 2 (3) developers, and contained fungi and protozoa in addition to the standard Kraken 2 (3) database (archaea, bacteria, viral, plasmid, human, UniVec_Core). 6B: The reference database used was a custom database (16 GB) curated by Kraken 2 (3) developers containing only fungal genomes. 6C: The reference database used for this classification was a custom database curated in-house with all fifty fungal species present in the FMC simulated dataset.Results were visualized using KronaTools (105).

7.3 FINDFUNGI

The implementation of the pipeline was unsuccessful. The 'Run_Statistics' file (**Fig. 7**) output by FindFungi (6) shows that the input file (the MMC simulated dataset) was read by the pipeline. The output 'Run_Statistics' file does not provide the number of reads for the subsequent steps in the pipeline.

```
Number of reads in the raw input:
1600000
Number of reads after trimming:
0
Number of reads removed by Kraken:
0
Number of reads remaining after Kraken:
0
Number of bacterial reads removed by BLAST:
0
Number of reads predicted as fungal:
```

Fig. 7. *Snapshot of the 'Run_Statistics' file output by FindFungi (6).* The implementation of the FindFungi (6) pipeline read the input MMC simulated dataset and output the number of reads in the dataset. The subsequent steps of the pipeline failed to execute.

The pipeline exited after echoing the number of reads in the raw input (**Fig. 8**). The next step, trimming low quality reads with Skewer (109), failed to run and the number of reads after trimming was given as zero (**Fig. 7**). The input file was successfully trimmed using Skewer (109) on the command line.

```
if [ ! -d $PreDir ]; then
        mkdir $PreDir
        echo "Number of reads in the raw input: " >> $PreDir/Run_Statistics.txt
        LinesInReadsIn=$(wc -l $x | awk '{print $1}')
        ReadsIn=$((LinesInReadsIn/4))
        echo $ReadsIn >> $PreDir/Run_Statistics.txt
        mkdir $PreDir/ReadTrimming
        bsub -K -q C skewer -l 30 -q 15 -t 30 -o $PreDir/ReadTrimming/$z $x &
        wait
        echo "Number of reads after trimming: " >> $PreDir/Run_Statistics.txt
        LinesInReadsLeft=$(wc -l $PreDir/ReadTrimming/$z-trimmed.fastq | awk '{print $1}')
        ReadsLeft=$((LinesInReadsLeft/4))
        echo $ReadsLeft >> $PreDir/Run_Statistics.txt
        mkdir $PreDir/FASTA
        sed -n '1~4s/^@/>/p;2~4p' $PreDir/ReadTrimming/$z-trimmed.fastq > $PreDir/FASTA/$z.fna #Convert FASTQ to FASTA
        LineCt=$(wc -l $PreDir/FASTA/$z.fna | awk '{print $1}')
        SplitN=$((LineCt/32 + 1))
        SplitI=$(printf "%.0f" $SplitN)
        split -l $SplitI $PreDir/FASTA/$z.fna $PreDir/FASTA/Split.
        for d in $PreDir/FASTA/*Split.*; do
                bsub -K -q C sed -i 's/\ /_/g' $d & #Replace whitespace with underscore
        done
```

Fig. 8. *Snapshot from the pipeline script 'FindFungi-0.23.3.sh'.* The implementation of the FindFungi (6) pipeline output only the number of reads in the input file.

7.3.1 Contacting the developers

The attempt to address the issue of the 'FindFungi-0.23.3.sh' script terminating after reading the raw input involved reaching out to the developers of FindFungi (6) for assistance. It was revealed that FindFungi (6) is no longer actively developed or maintained due to the developer's departure from the associated laboratory five years ago. As a result, the developer was unable to provide any further assistance. Any future updates to FindFungi (6) are unlikely.

7.3.2 FindFungi adapted for SLURM

A version of FindFungi (6) adapted for SLURM is available at https://github.com/astrophys/FindFungi_adapted_for_slurm. The SLURM adaptation of FindFungi (6) read the input file (the FMC simulated dataset) and removed low quality reads using Skewer (109). The 'Run_Statistics' file (**Fig. 9**) shows two reads were removed.

```
Number of reads in the raw input:
99260620
Number of reads after trimming:
99260618
```

Fig. 9. *A snapshot of the 'Run_Statistics' file output by FindFungi (6) adapted for SLURM.* Skewer (109) removed two reads from the FMC simulated dataset.

The subsequent steps of the pipeline failed to execute as the script exited with an sbatch error when attempting to run Kraken (104) (**Fig. 10**).

```
99260620 reads processed; of these:
       2 ( 0.00%) short reads filtered out after trimming by size control
       0 ( 0.00%) empty reads filtered out after trimming by size control
99260618 (100.00%) reads available; of these:
 2595723 ( 2.62%) trimmed reads available after processing
96664895 (97.38%) untrimmed reads available after processing
log has been saved to "FUNGALMC/ReadTrimming/FUNGALMC-trimmed.log".
Skewer ended: Sun Mar 26 14:56:03 CEST 2023
Sed starting : Sun Mar 26 14:59:57 CEST 2023
Sed ending : Sun Mar 26 15:00:52 CEST 2023
Starting : Kraken : Sun Mar 26 15:01:20 CEST 2023
sbatch: error: Account specification required, but not provided
sbatch: error: Batch job submission failed: Invalid account or account/partition combination specified
```

Fig. 10. *Snapshot from the SLURM job output file.* The script for FindFungi (6) adapted for SLURM exited after trimming reads with Skewer (109). Kraken (104) failed to run due to an sbatch error.

7.3.3 The original FindFungi script vs. FindFungi adapted for SLURM

The command bsub is used to submit a script to the job scheduler Load Sharing Facility (LSF). The comparison between the 'FindFungi-0.23.3.sh' script and the script used in FindFungi (6) adapted for SLURM reveals a notable distinction in the usage of bsub commands (**Fig. 11**). The former script employs bsub commands, imposing a memory limit on the executed commands. Conversely, the adapted version eliminates the use of bsub, thereby removing the memory constraint.

A

```
mkdir $PreDir/ReadTrimming
bsub -K -q C skewer -l 30 -q 15 -t 30 -o $PreDir/ReadTrimming/$z $x &
wait
echo "Number of reads after trimming: " >> $PreDir/Run_Statistics.txt
```

B

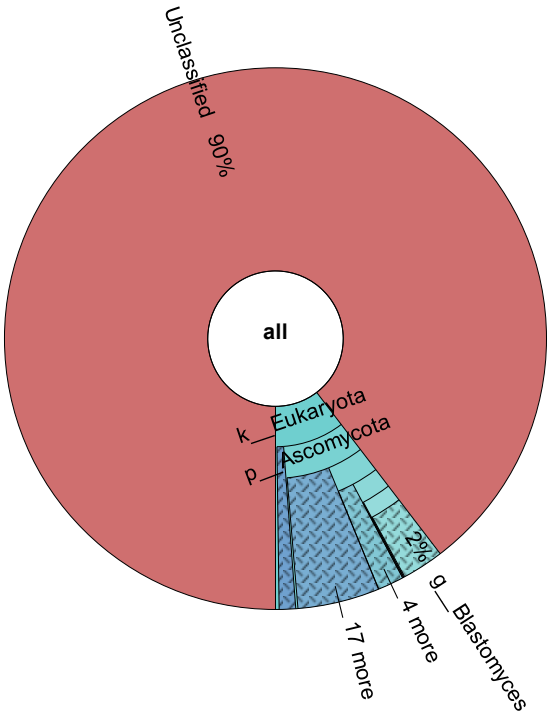
```
mkdir $PreDir/ReadTrimming

echo "Skewer starting : $(date)"
skewer -l 30 -q 15 -t ${NUM_THREADS} -o $PreDir/ReadTrimming/$z $x
echo "Skewer ended: $(date)"

echo "Number of reads after trimming: " >> $PreDir/Run_Statistics.txt
```

Fig. 11. *Snapshots from the two FindFungi (6) scripts.* 11A: The command used to execute Skewer (109) in the FindFungi-0.23.3.sh script. 11B: The command used to execute Skewer (109) in the FindFungi (6) SLURM adaptation.

7.3.4 Mapping the FMC simulated dataset against FindFungi's 32 Kraken databases with Snakemake

The FMC simulated dataset was mapped against the 32 Kraken (104) databases integrated into the FindFungi (6) pipeline using Snakemake (100). The Snakefile (**Appendix VI**) encompasses the necessary commands for mapping the FMC to the 32 Kraken (104) databases and consolidating the resulting 32 Kraken (104) reports into a single comprehensive report. A small portion (10.4 %) of the FMC simulated dataset was classified by the 32 Kraken (104) databases incorporated in the FindFungi (6) pipeline (**Fig. 12**).

A



B

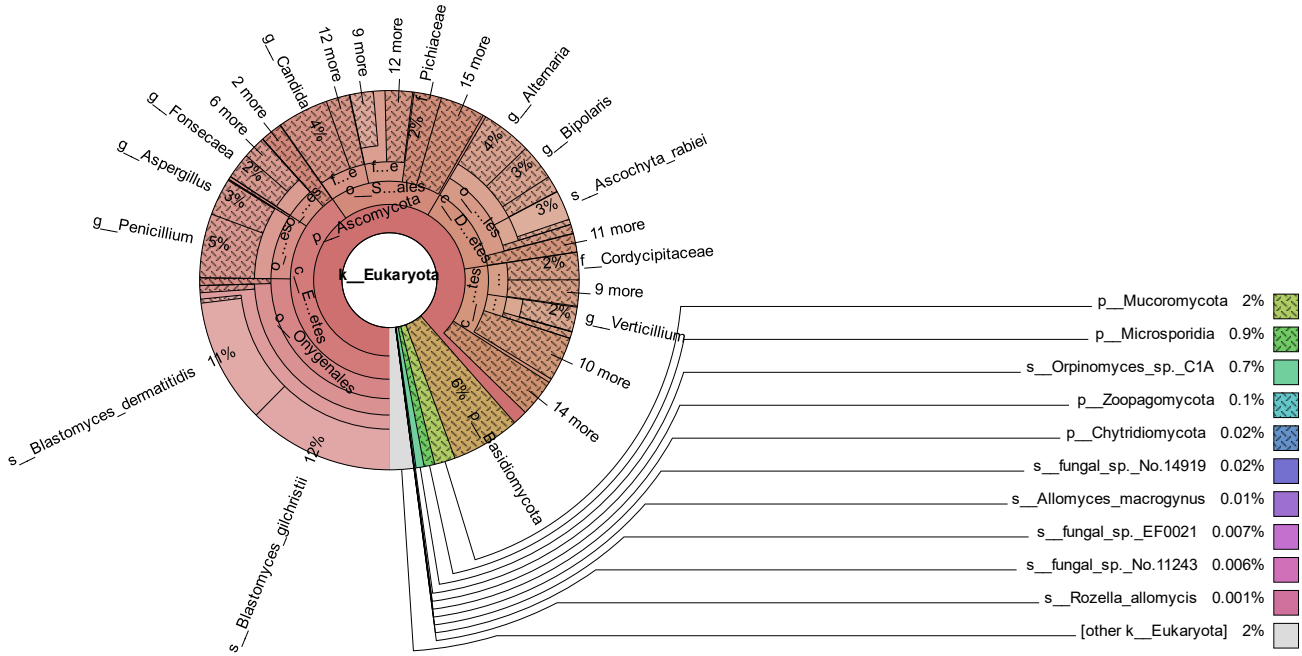


Fig. 12. *Mapping the fungal mock community (FMC) against FindFungi's (6) 32 Kraken (104) databases.* 12A: The proportion of classified and unclassified reads. 5B: The classified reads of the FMC. The reference databases used for this classification were the 32 Kraken (104) databases (16 GB each) incorporated in the FindFungi (6) pipeline. Results were visualized using KronaTools (105).

7.4 HUMANMYCOBIOMESCAN

During the attempt to execute the HumanMycobiomeScan (7) tool, it encountered failure and did not run successfully. The error message (**Fig. 13A**) indicated the absence of a required file for a successful run. Although the corresponding folder was created, the expected output file was not present in this folder. The command where the 'MScan.sh' script terminated (**Fig. 13B**) was identified. When contacting the developer for help, it was confirmed that the error was indeed due to the non-existence of the file. However, no specific explanation was provided for the cause of this occurrence. Insufficient time prevented conducting additional testing and evaluation of the software.

A

```
STARTING THE ANALYSIS
(ERR): Could not open output file '/cluster/projects/nn9383k/arfa/hms_analysis/HMS/mixmock//cluster/projects/nn9383k/arfa/hms_analysis/HMS/mixmock.sam' for writing.
Exiting now ...
```

B

```
echo -e 'STARTING THE ANALYSIS'

if [ "$INPUT_FASTQ_paired_end2" == "" ]
then
bowtie2 -x $HMS_PATH/database/bowtie2/$DATABASE -q $INPUT_FASTQ_paired_end1  --very-sensitive --no-unal -S $OUTPUT_DIR/$OUTPUT_DIR.sam -p $N_THREADS
else
bowtie2 -x $HMS_PATH/database/bowtie2/$DATABASE -1 $INPUT_FASTQ_paired_end1 -2 $INPUT_FASTQ_paired_end2 --very-sensitive --no-unal -S $OUTPUT_DIR/$OUTPUT_DIR.sa
fi

cat $OUTPUT_DIR/$OUTPUT_DIR.sam |grep -v ^@| awk '{print"@"$1"\n"$10"\n+\n"$11}' > $OUTPUT_DIR/$OUTPUT_DIR-funginofiltr.fastq
```

Fig. 13. *Troubleshooting the HumanMycobiomeScan (7) tool.* 13A: The error message encountered when running the HumanMycobiomeScan *(7)* tool indicated that it could not find the "mixmock.sam" file to start the analysis. 13B: A snapshot from the MScan.sh script shows the command HumanMycobiomeScan *(7)* was unable to execute.

7.5 FUNOMIC

The FunOMIC (5) pipeline encountered multiple challenges during implementation and was unable to run successfully. A number of error messages were encountered upon executing the pipeline. The taxonomic profiling log (**Fig. 14A**) indicated that the pipeline was unable to provide any output. Additionally, the functional profiling log reported an error related to the DIAMOND (107) software (**Fig. 14B**). The functional log revealed that DIAMOND (107) was unable to read the input file due to a seemingly blank first line in the input file. As a result, the functional profiling process was abruptly terminated. The first line of the input file was investigated, and the resulting output showed that the first line of the input file was not blank. Finally, the bacterial log revealed that the FunOMIC (5) pipeline script exited due to a bowtie2-align error (**Fig. 14C**).

A

```
0 reads
0.00% overall alignment rate
```

B

```
diamond v2.0.15.153 (C) Max Planck Society for the Advancement of Science
Documentation, support and updates available at http://www.diamondsearch.org
Please cite: http://dx.doi.org/10.1038/s41592-021-01101-x Nature Methods (2021)

#CPU threads: 80
Scoring parameters: (Matrix=BLOSUM62 Lambda=0.267 K=0.041 Penalties=11/1)
Temporary directory: /cluster/projects/nn9383k/arfa/funomic_analysis/output_folder/functional_profiling
#Target sequences to report alignments for: 25
Opening the database...  [0.447s]
Database: /cluster/projects/nn9383k/arfa/funomic_analysis/FunOMIC.P.v1//FunOMIC.P.v1.dmnd (type: Diamond database, sequences: 3413239, letters: 1879525586)
Block size = 5000000000
Opening the input file...  [0.001s]
Error: Error detecting input file format. First line seems to be blank.
```

C

```
(ERR): bowtie2-align died with signal 11 (SEGV) (core dumped)
```

Fig. 14. *The taxonomy, functional and bacterial decontamination logs output by FunOMIC (5).* 14A: The taxonomy log shows the pipeline was unable to read the input FMC simulated dataset and consequently did not align the dataset to reference genomes. 14B: The functional profiling log showing the error encountered by DIAMOND (107). 14C: An error was encountered with bowtie2-align, as presented in the bacterial decontamination log, that subsequently caused the script to exit.

Throughout the analysis process, encompassing taxonomic profiling, functional profiling, and the removal of bacterial reads, FunOMIC (5) consistently encountered difficulties in locating the required input file. This was evident in the generation of error messages at each step, as depicted in **Fig. 15**.

```
Start removing bacterial reads for testrun
[main_samview] fail to read the header from "/cluster/projects/nn9383k/arfa/funomic_analysis/output_folder/tmp/testrun_Bact.sam".
Start taxonomic annotation for testrun
[E::hts_open_format] Failed to open file "/cluster/projects/nn9383k/arfa/funomic_analysis/output_folder/tmp/testrun.sam" : No such file or directory
samtools view: failed to open "/cluster/projects/nn9383k/arfa/funomic_analysis/output_folder/tmp/testrun.sam" for reading: No such file or directory
[E::hts_open_format] Failed to open file "/cluster/projects/nn9383k/arfa/funomic_analysis/output_folder/tmp/testrun.sam" : No such file or directory
samtools view: failed to open "/cluster/projects/nn9383k/arfa/funomic_analysis/output_folder/tmp/testrun.sam" for reading: No such file or directory
[E::hts_open_format] Failed to open file "/cluster/projects/nn9383k/arfa/funomic_analysis/output_folder/tmp/testrun.30.sorted.bam" : No such file or directory
samtools view: failed to open "/cluster/projects/nn9383k/arfa/funomic_analysis/output_folder/tmp/testrun.30.sorted.bam" for reading: No such file or directory
grep: /cluster/projects/nn9383k/arfa/funomic_analysis/output_folder/tmp/testrun.30.c80.list: No such file or directory
[E::hts_open_format] Failed to open file "/cluster/projects/nn9383k/arfa/funomic_analysis/output_folder/tmp/testrun.30.c80.bam" : No such file or directory
samtools idxstats: failed to open "/cluster/projects/nn9383k/arfa/funomic_analysis/output_folder/tmp/testrun.30.c80.bam": No such file or directory
Starting functional annotation for testrun
cat: /cluster/projects/nn9383k/arfa/funomic_analysis/output_folder/functional_profiling/tmp/testrun.extendedFrags.fastq: No such file or directory
cat: /cluster/projects/nn9383k/arfa/funomic_analysis/output_folder/functional_profiling/tmp/testrun.notCombined_1.fastq: No such file or directory
cat: /cluster/projects/nn9383k/arfa/funomic_analysis/output_folder/functional_profiling/tmp/testrun.notCombined_2.fastq: No such file or directory
merged clean reads stored in: /cluster/projects/nn9383k/arfa/funomic_analysis/output_folder/functional_profiling/joined.fastq
awk: fatal: cannot open file `/cluster/projects/nn9383k/arfa/funomic_analysis/output_folder/functional_profiling/testrun.func.out' for reading (No such file or directory)
```

Fig. 15. *The SLURM job output of the FunOMIC (5) pipeline implementation.* The SLURM output showcases the error messages encountered during the execution of the FunOMIC (5) pipeline.

7.5.1 Communication and developer input: insights and contributions

The developer was contacted regarding the encountered issues during the execution of the FunOMIC (5) pipeline. In response, the developer acknowledged an error in the code and provided an updated version for further testing and implementation. However, upon execution of the new code, error messages were again encountered. The taxonomic profiling and bacterial decontamination logs remained the same as previously. The functional profiling log (**Fig. 16**) displayed a different error with the DIAMOND (107) software than the one encountered earlier.

```
diamond v2.0.15.153 (C) Max Planck Society for the Advancement of Science
Documentation, support and updates available at http://www.diamondsearch.org
Please cite: http://dx.doi.org/10.1038/s41592-021-01101-x Nature Methods (2021)

#CPU threads: 80
Scoring parameters: (Matrix=BLOSUM62 Lambda=0.267 K=0.041 Penalties=11/1)
Temporary directory: /cluster/projects/nn9383k/arfa/funomic_analysis/output_folder/functional_profiling
#Target sequences to report alignments for: 25
Opening the database...  [0.248s]
Database: /cluster/projects/nn9383k/arfa/funomic_analysis/FunOMIC.P.v1//FunOMIC.P.v1.dmnd (type: Diamond database, sequences: 3413239, letters: 1879525586)
Block size = 5000000000
Opening the input file... No such file or directory
 [0s]
Error: Error calling stat on file /cluster/projects/nn9383k/arfa/funomic_analysis/output_folder/functional_profiling/joined.fastq
```

Fig. 16. *The functional profiling log output by FunOMIC (5).* DIAMOND (107) encountered an error when opening the input file for the functional profiling of the FMC simulated dataset. The functional profiling log shows that no such file exists.

According to the developer, the encountered errors during functional profiling were attributed to issues with bacterial decontamination. It was highlighted that if the bacterial decontamination process is not successful, it will subsequently affect both taxonomic and functional profiling steps. Due to time constraints, it was not possible to conduct further testing and evaluation of the software.

8 DISCUSSION

8.1 KEY FINDINGS

In metagenomic analyses of microbiomes, and by extension the mycobiome, the initial step typically involves classifying reads based on their taxonomic information by comparing them to a database containing characterized genomes. Five bioinformatic tools, namely MetaPhlAn 3 (4), Kraken 2 (3), FindFungi (6), HumanMycobiomeScan (7) and FunOMIC (5), and their accompanying databases were evaluated for their ability to detect and classify fungal species in shotgun metagenomic datasets. Evaluation of the tools was conducted by mapping two simulated datasets against the databases of the tools. One simulated dataset, the mixed mock community (MMC), contained sixteen genomes (five bacterial, five fungal, five viral and the human genome). The second simulated dataset, the fungal mock community (FMC), contained fifty fungal genomes. The aim of this validation was to create an automated pipeline for mycobiome characterization by incorporating these tools into a Snakemake (100) pipeline. Rigorous benchmarking was performed to assess their effectiveness and reliability.

8.1.1 The impact of reference databases on Kraken 2 performance

A study conducted by Wright et. al found that the choice of reference database greatly impacts taxonomic classification (111). The FMC simulated dataset was mapped against three different custom Kraken 2 (3) databases to examine how the selection of a reference database impacts Kraken 2's (122) classification. Two of these databases, called PlusPF (available for download at: https://benlangmead.github.io/aws-indexes/k2) and FungiDB (this name is given for the purposes of this thesis and denotes the reference fungi library of Kraken 2 (3)) were curated by Kraken 2 (3) developers. Kraken 2 (3) classified 28.41 % and 25.53 % of the FMC simulated dataset when mapped against PlusPF and FungiDB, respectively. Despite the databases containing the same fungal genomes (both databases employ the same reference fungi library), using PlusPF as the reference database produced more classified reads than when using FungiDB. PlusPF contains archaeal, bacterial, viral, plasmid, human and a collection of known vectors (UniVec_Core) genomes in addition to fungal genomes. FungiDB on the other hand consists solely of fungal genomes. The inclusion of genomic content from other species producing a higher classification rate could be indicative of improved sensitivity and the ability to detect a broader range of taxonomic groups present in the sample. The correlation between the size and comprehensiveness of the reference

database and improved results is widely acknowledged, primarily to avoid inaccurate classifications caused by the omission of closely related organisms within a specific taxonomic group (112). This raises an important question about Kraken 2's (3) algorithm: *is it unable to detect fungal species accurately despite the presence of fungal genomes in reference databases?*

A custom Kraken 2 (3) database, called FMC_DB, was curated in-house to answer this question. The reference database encompassed all species present in the FMC simulated dataset. Kraken 2 (3) demonstrated the highest performance when mapping the FMC simulated dataset against the FMC_DB with a classification rate of 99.83 %. It is therefore logical to conclude that Kraken 2 (3) can indeed be employed for fungal classification of shotgun metagenomic datasets. Recent studies (23, 113, 114) employing Kraken 2 (3) to characterize the human gut mycobiome corroborate this conclusion.

A caveat to consider is that the effectiveness of Kraken 2 (122), and any classification tool in this instance, is limited when analyzing samples with novel species. The high performance of Kraken 2 (3) when employing FMC_DB shows that curating a custom database encompassing all species present in the sample allows it to accurately classify the sample. Constructing a sample-tailored database, however, is an impossible task when dealing with samples of unknown composition. To address this limitation and confidently classify all the sample reads, analyses would need to be conducted with a large enough reference database to ensure the inclusion of all possible organisms with classified genomes.

8.1.2 MetaPhlAn 3 limitations in fungal identification

MetaPhlAn 3 (4) exhibited limitations in accurately classifying all the fungal species present in both the MMC and FMC simulated datasets. The inability to classify certain species raises concerns about the comprehensiveness of MetaPhlan 3's (4) reference database in capturing the diversity of fungal taxa. This limitation could be attributed to the lack of eukaryotic reference genomes in the MetaPhlAn 3 (4) database. Of the approximately 99 200 reference genomes included in the MetaPhlan 3 (4) database, eukaryotic genomes account for only 0.12 % (or 122) of these (4). Consequently, analyses performed using MetaPhlAn 3's (4) provided database may lead to incomplete or inaccurate results, as researchers may fail to identify species that are actually present in a sample simply because they are not in MetaPhlAn 3's (4) provided database. This was shown to be the case for both the MMC and FMC simulated datasets where MetaPhlAn 3 (4) classified 60 % and 76 % of the fungal species present, respectively. Usyk et. al found an updated version of MetaPhlAn 3 (4), MetaPhlAn 4 (115),

was only able to detect fungi in 3.83 % of the samples compared to the ITS1 amplicon sequencing that identified fungi in 89.6 % of the samples (116). MetaPhlAn 4 (115) boasts a significant improvement in its database from MetaPhlAn 3 (4) with 169.1k reference genomes encompassing 31.9k species compared to the latter's 99.2 k genomes from 13.5 k species (115). Despite the expansion of the provided database in the latest update of MetaPhlAn, authors acknowledge that the present methods lack comprehensive integration of eukaryotic microbial sequences (115).

A literature search conducted on PubMed using the keywords "(MetaPhlAn 3) AND (mycobiome)" gave no hits whereas "(MetaPhlAn) AND (fungi)" returned nine hits. This indicates a gap in the current literature and highlights the need for further investigation and exploration of MetaPhlAn 3's (4) potential in studying fungal communities. Researchers relying solely on MetaPhlAn 3's (4) database should be cautious and consider alternative approaches or databases that encompass a broader range of species to ensure more comprehensive and accurate analysis of their samples.

8.1.3 Performance discrepancies and usability challenges of FindFungi

FindFungi (6), published in 2018, was developed with the aim of addressing the existing gap in the availability of tools specifically designed for fungal identification in shotgun metagenomic sequencing (6). This research objective was motivated by the recognition that at the time of publication, available tools in the field did not adequately capture the diversity and complexity of fungal communities present in such datasets. By curating a dedicated pipeline, FindFungi (6) sought to provide researchers with a valuable resource for accurate and efficient identification of fungal species. Donovan et. al tested five tools (BLAST (106), DIAMOND (107), Kaiju (108) and two versions of Kraken (104)) to determine the best method for classifying fungal reads from metagenomic datasets. Kraken (104) with the default *k*-mer setting of 31, was selected for the FindFungi (6) pipeline due to its speed, the amalgamation of excellent sensitivity and specificity, and its capability to assign a lowest common ancestor (LCA) prediction to every read (6). When a preliminary version of FindFungi (6) gave false positives, a read distribution step was added to the FindFungi (6) pipeline to prevent the occurrence of these. Classified reads from Kraken (104) are mapped to a simulated-assembly of the relevant genome using BLAST (106).

Despite the promising nature of the pipeline, the evaluation of FindFungi (6) revealed that the pipeline suffers from significant usability issues, making it cumbersome and not user-friendly. It became apparent that the tool's functionality and usability did not align with the

initial claims made by the authors. The setup process involves downloading multiple software and databases, as well as granting permissions to run the scripts, which can be challenging for users without a certain level of technical knowledge. Consequently, these requirements and complexities hinder the ease of getting started with FindFungi (6). Additionally, FindFungi (6) demands a substantial amount of storage space, further contributing to its impracticality. The need for ample storage capacity can pose challenges, especially for researchers with limited resources or those working with large datasets.

FindFungi's (6) script appears more tailored for project specific use rather than having universal applicability. One aspect that highlights this is the use of *bsub* commands in the script. While this approach may have been employed to optimize resource allocation, it inadvertently imposed restrictions on the types of tasks that could be performed. Consequently, several tools within the pipeline failed to execute their intended functions, leading to unforeseen limitations and reduced performance. One specific challenge faced was the incompatibility of the original FindFungi (6) script with the Skewer (109) tool. The input file was successfully trimmed using Skewer (109) outside of the FindFungi (6) pipeline. When a SLURM adapted version of FindFungi (6) was utilized, Skewer (109) was able to function as expected. This result suggests that the removal of *bsub* commands in the SLURM implementation of FindFungi (6) removes any memory restrictions and thus allows Skewer (109) to successfully run.

Though the implementation of FindFungi (6) remained unsuccessful, an attempt was made to classify the FMC simulated dataset with Snakemake (100) using FindFungi's (6) 32 Kraken (104) databases. Only 10.4 % of the FMC simulated dataset was classified. While Usyk et. al were able to recover more fungal species using FindFungi (6) than with ITS1 amplicon sequencing data; they concluded that the output obtained from FindFungi (6) is likely a consequence of incorrect categorization (116). Seen together with the low classification rate of the FMC simulated dataset, these results suggest that the reference databases used by FindFungi (6) to classify fungal reads are likely to be incomprehensive at best and wholly inaccurate at worst.

Another significant limitation encountered in this study is the lack of maintenance and updates for FindFungi (6). FindFungi (6) appears to be stagnant in terms of ongoing support and updates for the past five years. This limitation also raises concerns about the tool's compatibility with the latest advancements in fungal taxonomy and the availability of up-to-date reference databases. Without timely updates, FindFungi (6) may lack the inclusion of

newly discovered fungal species or taxonomic revisions, rendering it less reliable and comprehensive for fungal classification.

8.1.4 Discrepancies, technical challenges, and the need for benchmarking of HumanMycobiomeScan and FunOMIC

HumanMycobiomeScan (7) and FunOMIC (5) are two tools that have shown potential for analyzing fungal communities in metagenomic studies. However, a closer examination of these tools reveals several limitations that need to be addressed before their widespread adoption in the field.

One of the primary limitations is the discrepancies between the information presented in the published papers and the actual implementation of the provided code. Both HumanMycobiomeScan (7) and FunOMIC (5) suffered from technical issues, with scripts frequently generating error messages. The lack of proper error handling and troubleshooting support, particularly in the case of HumanMycobiomeScan (7), further compounded the difficulties faced by users.

Moreover, neither of the tools have undergone rigorous benchmarking to evaluate their performance in diverse scenarios and against established standards. FunOMIC (5) has only recently been published (October 2022). At the time of writing in May 2023, FunOMIC (5) has one single citation from March 2023. Notably, this citation was made by one of the authors of FunOMIC (5), suggesting that the tool has yet to gain widespread recognition or uptake by other researchers in the field. Similarly, HumanMycobiomeScan (7), since its publication in June of 2019, has had few citations adopting the tool. To establish the credibility and utility of these tools, thorough benchmarking studies should be conducted to assess their performance compared to established approaches. This would involve evaluating their accuracy, precision, recall, and computational efficiency across a range of datasets and experimental conditions.

8.2 STRENGTHS AND LIMITATIONS

Unlike targeted approaches such as amplicon sequencing that focus on specific regions like ITS (6), shotgun metagenomics allows for the unbiased sequencing of entire microbiomes, providing a more comprehensive view of fungal communities, such as the mycobiome. This approach enables the identification of not only known fungi but also novel and rare species that may have been missed using targeted methods (86). A decrease in sequencing costs has given researchers greater access to HTS technologies. Informatics software development is

progressing swiftly and improving the ease and effectiveness of metagenomic analysis (86). Some challenges remain that hinder widespread adoption of shotgun metagenomic sequencing in mycobiome characterization.

Firstly, there is a clear lack of available literature on human gut mycobiome research using shotgun metagenomic sequencing data. PubMed gives eight results when using the search query "(shotgun metagenomic sequencing) AND (human gut mycobiome)". The articles are all published within the past four years (the earliest in April 2019). One study used fecal shotgun metagenomic sequences CRC patients, individuals with adenomas and healthy controls in Hong Kong to characterize the enteric mycobiome in CRC (75). The study uncovered fungal dysbiosis in the gut specifically related to CRC (75). This imbalance in the mycobiome, marked by changes in fungal composition and ecological patterns, suggests a potential involvement of the gut fungal community in the development of CRC (75).

In the past, the primary emphasis of microbiome research was on the bacteriome, resulting in a lack of standardization in techniques for studying the mycobiome in the gut (117). Consequently, the progress of these studies is hindered by the absence of standardized protocols, technical challenges, a scarcity of reference data, and potential biases in data analysis (28).

Secondly, the availability of reference fungal genomes in reference databases that accompany current classification tools is insufficient for conducting fungal classification studies. Every metagenomics computational tool depends, to some degree, on the accessibility of reference genomes, which means that any prejudices present in the reference sequence resources can have an impact on them (99). Additionally, shotgun metagenomics sequencing has not yet achieved the degree of standardization that is typical of other more established HTS methods (99). The substantial proportion of fungal species from the two simulated datasets that could not be classified during the assessments of Kraken 2 (3) and MetaPhlAn 3 (4) as well as FindFungi's (6) reference databases highlights the existing constraints of the currently accessible databases.

Finally, software that is not regularly updated and maintained is a limitation of this thesis. FindFungi (6) is not actively maintained and has not received any updates to its code since publication in 2018. The absence of user support exacerbated the difficulties associated with debugging code and addressing problems related to reproducibility. Ensuring the integrity and quality of bioinformatics pipelines is crucial for delivering reproducible and high-quality outcomes. Neglecting to regularly update the code may lead to getting trapped in repetitive troubleshooting cycles, as was the case with the implementation of FindFungi (6).

8.3 FUTURE CONSIDERATIONS

Fungal genomes exhibit a high level of diversity and complexity compared to bacterial genomes. They often possess a variable number of chromosomes, which can complicate the process of relative quantification when using certain tools like Kraken 2 (3). Kraken 2 (3) quantifies the abundance of different taxa by counting the number of reads assigned to each taxonomic group. However, Kraken 2 (3) does not account for variations in genome size when estimating abundance. Without considering genome size, the abundance estimates obtained may not accurately reflect the true proportions of fungal taxa present in a given sample. In contrast, tools like MetaPhlAn 3 (4) take into consideration the genome size of different taxa when performing relative quantification. By normalizing the abundance estimates by genome size, MetaPhlAn 3 (4) provides a more accurate representation of the relative abundance of different microbial species, including fungi. Any future tools developed for fungal classification should consider the diverse nature of fungal genomes when curating reference databases to obtain more reliable and informative results.

As mentioned previously under limitations, the lack of fungal reference genomes severely impacts the performance of classification tools. Future work surrounding a pipeline for mycobiome characterization would require the construction of a sizable and comprehensive reference database to ensure adequate classification of fungal species.

8.3.1 Constructing a Snakemake pipeline for fungal identification in shotgun metagenomic datasets

Due to time constraints and challenges faced during evaluation of the classification tools, the aim of developing a Snakemake (100) workflow was not achieved. Suggestions for an optimized Snakemake pipeline (100) for fungal identification in shotgun metagenomic datasets, based on findings in this thesis, are presented below.

The process of pipeline development typically involves establishing an infrastructure, constructing a computational workflow, and examining the obtained outcomes. Given the prevalence of sequencing technology and bioinformatics analysis, it is important to design a pipeline that minimizes the need for extensive computational or coding knowledge. Ideally, the pipeline should facilitate analysis through a user-friendly graphical user interface (GUI) that allows for straightforward point-and-click operations.

Before incorporating any software or tools within the pipeline, the software/tools should be tested in a new, unconfigured environment to ensure that results can be reproduced. Additionally, the number of dependencies that require installation before the pipeline is

operational should be minimized. By incorporating the installation of the necessary software within the pipeline code (as opposed to requiring users to install them separately), users can experience a less burdensome installation process. This also mitigates the risk of user error during installation and avoids implementation failure. For any dependencies needed, the known working version should be included in the documentation of the pipeline.

The pipeline should contain the following steps: (i) pre-processing of reads, (ii) mapping reads to reference genomes, (iii) taxonomic classification, and (iv) functional annotation. This provides users with a full-fledged metagenomic analysis of their sample from start to finish.

The reference database employed by the pipeline should comprehensively comprise genomes of both fungal and non-fungal origin. Evidence suggests using reference databases that consist of a single taxonomic group results in an unacceptably elevated rate of false-positive findings, primarily due to two factors: (i) mapping to conserved genetic regions in reference genomes, and (ii) contamination of sequences in the assembled reference genomes (112). To account for the diversity of fungal genomes, the reference database should employ marker genes similar to that of MetaPhlAn 3 (4). To meet the needs of the users' analyses, an option to curate a custom database should also be included.

Finally, the pipeline should be regularly maintained and receive careful attention to updates to the code. Additionally, user support should be readily available to facilitate any troubleshooting. The rapid developments in the field of microbiome (and mycobiome, albeit less rapid) research requires constant updates and maintenance of software. This is a vital aspect of software development and developers should ensure allocation of necessary resources for the upkeep of any published pipeline to facilitate universal adoption.

9 CONCLUSION

Fungi exhibit a vast array of morphological and ecological variations. This poses difficulties in their characterization due to the inherent challenges associated with culturing them. Culture-independent methods such as shotgun metagenomic sequencing bypass this limitation. This thesis aimed to evaluate and compare five bioinformatic tools (MetaPhlAn 3 (4), Kraken 2 (3), FindFungi (6), HumanMycobiomeScan (7), and FunOMIC (5)) for their ability to detect and classify fungal species in shotgun metagenomic datasets. The findings of this thesis shed light on the strengths and limitations of these tools and provide valuable insights for researchers interested in studying the mycobiome.

While HumanMycobiomeScan (7) and FunOMIC (5) showed potential for analyzing fungal communities, they also suffered from discrepancies between the published papers and the actual implementation of the provided code. Technical issues and a lack of rigorous benchmarking studies hinder their widespread adoption and raise concerns about their accuracy and performance compared to established approaches.

FindFungi (6), despite its initial promise of addressing the gap in fungal identification tools, suffered from usability challenges and limitations. The setup process, including downloading multiple software and databases, and the demand for ample storage space makes the tool cumbersome and impractical for users with limited technical knowledge of command-line and shell scripts. The tool's script appeared more tailored for project-specific use, and its compatibility with other software and tools was limited as a consequence. The reference databases used by FindFungi (6) for classification of fungal reads were also found to be incomprehensive and inaccurate. Lack of maintenance and updates further raised concerns about the tool's reliability and compatibility with the latest advancements in fungal taxonomy.

Kraken 2 (3) and MetaPhlAn 3 (4) are two well-established classification tools with hundreds of citations each, but the use of these tools for fungal classification is not without challenges. It was found that the choice of reference database greatly affects the tools' classification accuracy. The limited number of eukaryotic genomes included in the provided databases raises concerns about the comprehensiveness of capturing the diversity of fungal taxa. The inclusion of genomic content from other species in the reference database, in addition to fungal genomes, improved the classification rate, indicating the importance of comprehensive and diverse reference databases for accurate classification.

This thesis has highlighted the strengths and limitations of different bioinformatics tools for studying the mycobiome using shotgun metagenomic sequencing. The unbiased nature of shotgun metagenomics provides a comprehensive view of fungal communities, enabling the identification of known and novel species. However, challenges such as technical difficulties and lack of reference data still need to be addressed. Future research should focus on developing standardized approaches, improving reference databases, and conducting rigorous benchmarking studies to advance our understanding of the mycobiome and its role in human health and disease.

10 BIBLIOGRAPHY

1.Thursby E, Juge N. Introduction to the human gut microbiota. Biochem J. 2017;474(11):1823-36.
2.Matijasic M, Mestrovic T, Paljetak HC, Peric M, Baresic A, Verbanac D. Gut Microbiota beyond Bacteria-Mycobiome, Virome, Archaeome, and Eukaryotic Parasites in IBD. Int J Mol Sci. 2020;21(8):2668.
3.Wood DE, Lu J, Langmead B. Improved metagenomic analysis with Kraken 2. Genome Biology. 2019;20(1):1-257.
4.Beghini F, McIver LJ, Blanco-Miguez A, Dubois L, Asnicar F, Maharjan S, et al. Integrating taxonomic, functional, and strain-level profiling of diverse microbial communities with bioBakery 3. Elife. 2021;10.
5.Xie Z, Manichanh C. FunOMIC: Pipeline with built-in fungal taxonomic and functional databases for human mycobiome profiling. Computational and structural biotechnology journal. 2022;20:3685-94.
6.Donovan PD, Gonzalez G, Higgins DG, Butler G, Ito K. Identification of fungi in shotgun metagenomics datasets. PLoS One. 2018;13(2):e0192898.
7.Soverini M, Turroni S, Biagi E, Brigidi P, Candela M, Rampelli S. HumanMycobiomeScan: a new bioinformatics tool for the characterization of the fungal fraction in metagenomic samples. BMC Genomics. 2019;20(1):496-.
8.Guinane CM, Cotter PD. Role of the gut microbiota in health and chronic gastrointestinal disease: understanding a hidden metabolic organ. Therap Adv Gastroenterol. 2013;6(4):295-308.
9.Li J, Jia H, Cai X, Zhong H, Feng Q, Sunagawa S, et al. An integrated catalog of reference genes in the human gut microbiome. Nat Biotechnol. 2014;32(8):834-41.
10.Qin J, Li R, Raes J, Arumugam M, Burgdorf KS, Manichanh C, et al. A human gut microbial gene catalogue established by metagenomic sequencing. Nature (London). 2010;464(7285):59-65.
11.Turnbaugh PJ, Ley RE, Hamady M, Fraser-Liggett CM, Knight R, Gordon JI. The Human Microbiome Project. Nature. 2007;449(7164):804-10.
12.Itai S, Narciso Martín Q, Pasolli E, Fabbrini M, Vitali F, Agamennone V, et al. The Core Human Microbiome: Does It Exist and How Can We Find It? A Critical Review of the Concept. Nutrients. 2022;14(14):2872.
13.Gill SR, Pop M, DeBoy RT, Eckburg PB, Turnbaugh PJ, Samuel BS, et al. Metagenomic Analysis of the Human Distal Gut Microbiome. Science. 2006;312(5778):1355-9.
14.Lozupone CA, Stombaugh JI, Gordon JI, Jansson JK, Knight R. Diversity, stability and resilience of the human gut microbiota. Nature. 2012;489(7415):220-30.
15.Natividad JMM, Verdu EF. Modulation of intestinal barrier by intestinal microbiota: Pathological and therapeutic implications. Pharmacol Res. 2013;69(1):42-51.
16.Bäumler AJ, Sperandio V. Interactions between the microbiota and pathogenic bacteria in the gut. Nature. 2016;535(7610):85-93.
17.Gensollen T, Iyer SS, Kasper DL, Blumberg RS. How colonization by microbiota in early life shapes the immune system. Science. 2016;352(6285):539-44.
18.Gomaa EZ. Human gut microbiota/microbiome in health and diseases: a review. Antonie van Leeuwenhoek. 2020;113(12):2019-40.
19.Shin W, Kim HJ. Intestinal barrier dysfunction orchestrates the onset of inflammatory host–microbiome cross-talk in a human gut inflammation-on-a-chip. Proc Natl Acad Sci U S A. 2018;115(45):E10539-E47.

20.Levescot A, Malamut G, Cerf-Bensussan N. Immunopathogenesis and environmental triggers in coeliac disease. Gut. 2022;71(11):2337-49.
21.Cerf-Bensussan N, Gaboriau-Routhiau V. The immune system and the gut microbiota: friends or foes? Nat Rev Immunol. 2010;10(10):735-44.
22.Qin X, Gu Y, Liu T, Wang C, Zhong W, Wang B, et al. Gut mycobiome: A promising target for colorectal cancer. Biochim Biophys Acta Rev Cancer. 2021;1875(1):188489.
23.Hu Y, Irinyi L, Hoang MTV, Eenjes T, Graetz A, Stone EA, et al. Inferring Species Compositions of Complex Fungal Communities from Long- and Short-Read Sequence Data. mBio. 2022;13(2):e0244421.
24.Ghannoum MA, Jurevic RJ, Mukherjee PK, Cui F, Sikaroodi M, Naqvi A, et al. Characterization of the oral fungal microbiome (mycobiome) in healthy individuals. PLoS Pathog. 2010;6(1):e1000713.
25.Zhang F, Aschenbrenner D, Yoo JY, Zuo T. The gut mycobiome in health, disease, and clinical applications in association with the gut bacterial microbiome assembly. Lancet Microbe. 2022;3(12):e969-e83.
26.Hoffmann C, Dollive S, Grunberg S, Chen J, Li H, Wu GD, et al. Archaea and fungi of the human gut microbiome: correlations with diet and bacterial residents. PLoS One. 2013;8(6):e66019-e.
27.Hallen-Adams HE, Suhr MJ. Fungi in the healthy human gastrointestinal tract. Virulence. 2017;8(3):352-8.
28.Suhr MJ, Hallen-Adams HE. The human gut mycobiome: pitfalls and potentials-a mycologist's perspective. Mycologia. 2015;107(6):1057-73.
29.Mar Rodríguez M, Pérez D, Javier Chaves F, Esteve E, Marin-Garcia P, Xifra G, et al. Obesity changes the human gut mycobiome. Sci Rep. 2015;5(1):14600-.
30.Scanlan PD, Marchesi JR. Micro-eukaryotic diversity of the human distal gut microbiota: qualitative assessment using culture-dependent and -independent analysis of faeces. ISME J. 2008;2(12):1183-93.
31.Hallen-Adams HE, Kachman SD, Kim J, Legge RM, Martínez I. Fungi inhabiting the healthy human gastrointestinal tract: a diverse and dynamic community. Fungal ecology. 2015;15:9-17.
32.Laforest-Lapointe I, Arrieta M-C. Patterns of Early-Life Gut Microbial Colonization during Human Immune Development: An Ecological Perspective. Front Immunol. 2017;8:788-.
33.Fujimura KEKE, Sitarik ARAR, Havstad SS, Lin DLDL, Levan SS, Fadrosh DD, et al. Neonatal gut microbiota associates with childhood multi–sensitized atopy and T–cell differentiation. Nature medicine. 2016;22(10):1187-91.
34.Iliev ID, Leonardi I. Fungal dysbiosis: immunity and interactions at mucosal barriers. Nat Rev Immunol. 2017;17(10):635-46.
35.Gu Y, Zhou G, Qin X, Huang S, Wang B, Cao H. The Potential Role of Gut Mycobiome in Irritable Bowel Syndrome. Front Microbiol. 2019;10:1894-.
36.Liguori G, Lamas B, Richard ML, Brandi G, da Costa G, Hoffmann TW, et al. Fungal Dysbiosis in Mucosa-associated Microbiota of Crohn's Disease Patients. J Crohns Colitis. 2016;10(3):296-305.
37.Strati F, Cavalieri D, Albanese D, De Felice C, Donati C, Hayek J, et al. New evidences on the altered gut microbiota in autism spectrum disorders. Microbiome. 2017;5(1):24-.
38.Severance EG, Alaedini A, Yang S, Halling M, Gressitt KL, Stallings CR, et al. Gastrointestinal inflammation and associated immune activation in schizophrenia. Schizophr Res. 2012;138(1):48-53.
39.Huseyin CE, O'Toole PW, Cotter PD, Scanlan PD. Forgotten fungi—the gut mycobiome in human health and disease. FEMS Microbiol Rev. 2017;41(4):479-511.

40.Chu H, Duan Y, Lang S, Jiang L, Wang Y, Llorente C, et al. The Candida albicans exotoxin candidalysin promotes alcohol-associated liver disease. J Hepatol. 2020;72(3):391-400.
41.Van Dyken SJ, Garcia D, Porter P, Huang X, Quinlan PJ, Blanc PD, et al. Fungal chitin from asthma-associated home environments induces eosinophilic lung infiltration. J Immunol. 2011;187(5):2261-7.
42.Noverr MC, Falkowski NR, McDonald RA, McKenzie AN, Huffnagle GB. Development of Allergic Airway Disease in Mice following Antibiotic Therapy and Fungal Microbiota Increase: Role of Host Genetics, Antigen, and Interleukin-13. Infect Immun. 2005;73(1):30-8.
43.Zuo T, Zhan H, Zhang F, Liu Q, Tso EYK, Lui GCY, et al. Alterations in Fecal Fungal Microbiome of Patients With COVID-19 During Time of Hospitalization until Discharge. Gastroenterology. 2020;159(4):1302-10.e5.
44.Noverr MC, Phare SM, Toews GB, Coffey MJ, Huffnagle GB. Pathogenic Yeasts Cryptococcus neoformans and Candida albicans Produce Immunomodulatory Prostaglandins. Infect Immun. 2001;69(5):2957-63.
45.Perfect JR, Casadevall A. Fungal molecular pathogenesis: what can it do and why do we need it? In Molecular principles of fungal pathogenesis. Washington, DC: ASM Press; 2006.
46.Morgan E, Arnold M, Gini A, Lorenzoni V, Cabasag CJ, Laversanne M, et al. Global burden of colorectal cancer in 2020 and 2040: incidence and mortality estimates from GLOBOCAN. Gut. 2022;72(2):338-44.
47.Sung H, Ferlay J, Siegel RL, Laversanne M, Soerjomataram I, Jemal A, et al. Global Cancer Statistics 2020: GLOBOCAN Estimates of Incidence and Mortality Worldwide for 36 Cancers in 185 Countries. CA Cancer J Clin. 2021;71(3):209-49.
48.Norway CRo. Cancer in Norway 2021 - Cancer incidence, mortality, survival and prevalence in Norway. 2022 08 June, 2022.
49.Observatory GC. Cancer Today: International Agency for Research on Cancer; 2020 [
50.Dekker E, Tanis PJ, Vleugels JLA, Kasi PM, Wallace MB. Colorectal cancer. The Lancet (British edition). 2019;394(10207):1467-80.
51.Chen L, Ye L, Hu B. Hereditary Colorectal Cancer Syndromes: Molecular Genetics and Precision Medicine. Biomedicines. 2022;10(12):3207.
52.Botteri E, Iodice S, Bagnardi V, Raimondi S, Lowenfels AB, Maisonneuve P. Smoking and Colorectal Cancer: A Meta-analysis. JAMA. 2008;300(23):2765-78.
53.Cai S, Li Y, Ding Y, Chen K, Jin M. Alcohol drinking and the risk of colorectal cancer death: a meta-analysis. Eur J Cancer Prev. 2014;23(6):532-9.
54.Kyrgiou M, Kalliala I, Markozannes G, Gunter MJ, Paraskevaidis E, Gabra H, et al. Adiposity and cancer at major anatomical sites: umbrella review of the literature. BMJ. 2017;356:j477-j.
55.Kværner AS, Birkeland E, Bucher-Johannessen C, Vinberg E, Nordby JI, Kangas H, et al. The CRCbiome study : a large prospective cohort study examining the role of lifestyle and the gut microbiome in colorectal cancer screening participants. BMC cancer. 2021;21(1):1-930.
56.Bae JM, Kim JH, Kang GH. Molecular subtypes of colorectal cancer and their clinicopathologic features, with an emphasis on the serrated neoplasia pathway. Arch Pathol Lab Med. 2016;140(5):406-12.
57.Fearon ER, Vogelstein B. A genetic model for colorectal tumorigenesis. Cell. 1990;61(5):759-67.
58.Torgovnick A, Schumacher B. DNA repair mechanisms in cancer development and therapy. Front Genet. 2015;6:157-.

59.Barbacid M. ras Genes. Annu Rev Biochem. 1987;56(1):779-827.
60.Damin F, Galbiati S, Soriani N, Burgio V, Ronzoni M, Ferrari M, et al. Analysis of KRAS, NRAS and BRAF mutational profile by combination of in-tube hybridization and universal tag-microarray in tumor tissue and plasma of colorectal cancer patients. PLoS One. 2018;13(12):e0207876-e.
61.Dienstmann R, Connor K, Byrne AT, Fridman WH, Lambrechts D, Sadanandam A, et al. Precision Therapy in RAS Mutant Colorectal Cancer. Gastroenterology. 2020;158(4):806-11.
62.Grady WM, Lao VV. Epigenetics and colorectal cancer. Nat Rev Gastroenterol Hepatol. 2011;8(12):686-700.
63.Jardé T, Evans RJ, McQuillan KL, Parry L, Feng GJ, Alvares B, et al. In vivo and in vitro models for the therapeutic targeting of Wnt signaling using a Tet-OΔN89β-catenin system. Oncogene. 2013;32(7):883-93.
64.Krasinskas AM. EGFR Signaling in Colorectal Carcinoma. Patholog Res Int. 2011;2011:932932-6.
65.Labianca R, Nordlinger B, Beretta GD, Mosconi S, Mandalà M, Cervantes A, et al. Early colon cancer: ESMO Clinical Practice Guidelines for diagnosis, treatment and follow-up. Ann Oncol. 2013;24 Suppl 6:vi64-vi72.
66.Corley DA, Jensen CD, Marks AR, Zhao WK, Lee JK, Doubeni CA, et al. Adenoma Detection Rate and Risk of Colorectal Cancer and Death. N Engl J Med. 2014;370(14):1298-306.
67.Clancy C, O'Leary DP, Burke JP, Redmond HP, Coffey JC, Kerin MJ, et al. A meta-analysis to determine the oncological implications of conversion in laparoscopic colorectal cancer surgery. Colorectal Dis. 2015;17(6):482-90.
68.Atkin WP, Wooldrage KM, Parkin DMMD, Kralj-Hans IP, MacRae EP, Shah UM, et al. Long term effects of once-only flexible sigmoidoscopy screening after 17 years of follow-up: the UK Flexible Sigmoidoscopy Screening randomised controlled trial. Lancet. 2017;389(10076):1299-311.
69.Schoen RE, Pinsky PF, Weissfeld JL, Yokochi LA, Church T, Laiyemo AO, et al. Colorectal-Cancer Incidence and Mortality with Screening Flexible Sigmoidoscopy. N Engl J Med. 2012;366(25):2345-57.
70.Holme Ø, Løberg M, Kalager M, Bretthauer M, Hernán MA, Aas E, et al. Effect of Flexible Sigmoidoscopy Screening on Colorectal Cancer Incidence and Mortality: A Randomized Clinical Trial. JAMA. 2014;312(6):606-15.
71.Lindholm E, Brevinge H, Haglind E. Survival benefit in a randomized clinical trial of faecal occult blood screening for colorectal cancer. Br J Surg. 2008;95(8):1029-36.
72.Randel KR, Schult AL, Botteri E, Hoff G, Bretthauer M, Ursin G, et al. Colorectal Cancer Screening With Repeated Fecal Immunochemical Test Versus Sigmoidoscopy: Baseline Results From a Randomized Trial. Gastroenterology. 2021;160(4):1085-96.e5.
73.Bailey SER, Abel GA, Atkins A, Byford R, Davies S-J, Mays J, et al. Diagnostic performance of a faecal immunochemical test for patients with low-risk symptoms of colorectal cancer in primary care: an evaluation in the South West of England. Br J Cancer. 2021;124(7):1231-6.
74.Segata N, Waldron LD, Ballarini A, Narasimhan V, Jousson O, Huttenhower C. Metagenomic microbial community profiling using unique clade-specific marker genes. 2012.
75.Coker OO, Nakatsu G, Dai RZ, Wu WKK, Wong SH, Ng SC, et al. Enteric fungal microbiota dysbiosis and ecological alterations in colorectal cancer. Gut. 2019;68(4):654-62.
76.Luan C, Xie L, Yang X, Miao H, Lv N, Zhang R, et al. Dysbiosis of fungal microbiota in the intestinal mucosa of patients with colorectal adenomas. Sci Rep. 2015;5(1):7980-.

77.Gao R, Kong C, Li H, Huang L, Qu X, Qin N, et al. Dysbiosis signature of mycobiota in colon polyp and colorectal cancer. Eur J Clin Microbiol Infect Dis. 2017;36(12):2457-68.
78.Sokol H, Leducq V, Aschard H, Pham H-P, Jegou S, Landman C, et al. Fungal microbiota dysbiosis in IBD. Gut. 2017;66(6):1039-48.
79.Ramirez-Garcia A, Rementeria A, Aguirre-Urizar JM, Moragues MD, Antoran A, Pellon A, et al. Candida albicans and cancer: Can this yeast induce cancer development or progression? Crit Rev Microbiol. 2016;42(2):181-93.
80.Chin S-F, Megat Mohd Azlan PIH, Mazlan L, Neoh H-M. Identification of Schizosaccharomyces pombe in the guts of healthy individuals and patients with colorectal cancer: preliminary evidence from a gut microbiome secretome study. Gut Pathog. 2018;10(1):29-.
81.Lindahl BD, Nilsson RH, Tedersoo L, Abarenkov K, Carlsen T, Kjøller R, et al. Fungal community analysis by high-throughput sequencing of amplified markers – a user's guide. New Phytol. 2013;199(1):288-99.
82.Hoggard M, Vesty A, Wong G, Montgomery JM, Fourie C, Douglas RG, et al. Characterizing the Human Mycobiota: A Comparison of Small Subunit rRNA, ITS1, ITS2, and Large Subunit rRNA Genomic Targets. Front Microbiol. 2018;9:2208-.
83.Hamady M, Knight R. Microbial community profiling for human microbiome projects: Tools, techniques, and challenges. Genome Res. 2009;19(7):1141-52.
84.Hong S, Bunge J, Leslin C, Jeon S, Epstein SS. Polymerase chain reaction primers miss half of rRNA microbial diversity. ISME J. 2009;3(12):1365-73.
85.Sharpton TJ, Riesenfeld SJ, Kembel SW, Ladau J, O'Dwyer JP, Green JL, et al. PhyLOTU: A high-throughput procedure quantifies microbial community diversity and resolves novel taxa from metagenomic data. PLoS Comput Biol. 2011;7(1):e1001061-e.
86.Sharpton TJ. An introduction to the analysis of shotgun metagenomic data. Front Plant Sci. 2014;5:209-.
87.Quince C, Walker AW, Simpson JT, Loman NJ, Segata N. Shotgun metagenomics, from sampling to analysis. Nat Biotechnol. 2017;35(9):833-44.
88.Yuan S, Cohen DB, Ravel J, Abdo Z, Forney LJ. Evaluation of methods for the extraction and purification of DNA from the human microbiome. PLoS One. 2012;7(3):e33865-e.
89.Wesolowska-Andersen A, Bahl MI, Carvalho V, Kristiansen K, Sicheritz-Pontén T, Gupta R, et al. Choice of bacterial DNA extraction method from fecal material influences community structure as evaluated by metagenomic analysis. Microbiome. 2014;2(1):19-.
90.Amarasinghe SL, Su S, Dong X, Zappia L, Ritchie ME, Gouil Q. Opportunities and challenges in long-read sequencing data analysis. Genome Biol. 2020;21(1):30-.
91.Luo C, Tsementzi D, Kyrpides NC, Konstantinidis KT. Individual genome assembly from complex community short-read metagenomic datasets. ISME J. 2012;6(4):898-901.
92.Freedman AH, Gaspar JM, Sackton TB. Short paired-end reads trump long single-end reads for expression analysis. BMC Bioinformatics. 2020;21(1):149-.
93.Nilsson RH, Anslan S, Bahram M, Wurzbacher C, Baldrian P, Tedersoo L. Mycobiome diversity: high-throughput sequencing and identification of fungi. Nat Rev Microbiol. 2019;17(2):95-109.
94.Cock PJA, Fields CJ, Goto N, Heuer ML, Rice PM. Sanger FASTQ file format for sequences with quality scores, and the Solexa/Illumina FASTQ variants. Nucleic Acids Res. 2010;38(6):1767-71.
95.Ghurye JS, Cepeda-Espinoza V, Pop M. Metagenomic Assembly: Overview, Challenges and Applications. Yale J Biol Med. 2016;89(3):353-62.
96.Grigoriev IV, Nordberg H, Shabalov I, Aerts A, Cantor M, Goodstein D, et al. The Genome Portal of the Department of Energy Joint Genome Institute. Nucleic Acids Res. 2012;40(D1):D26-D32.

97.Tamames J, Cobo-Simón M, Puente-Sánchez F. Assessing the performance of different approaches for functional and taxonomic annotation of metagenomes. BMC Genomics. 2019;20(1):960-.
98.Huson DH, Mitra S, Ruscheweyh H-J, Weber N, Schuster SC. Integrative analysis of environmental sequences using MEGAN4. Genome Res. 2011;21(9):1552-60.
99.Vuong P, Wise MJ, Whiteley AS, Kaur P. Ten simple rules for investigating (meta)genomic data from environmental ecosystems. PLoS Comput Biol. 2022;18(12):e1010675-e.
100.Mölder F, Jablonski KP, Letcher B, Hall MB, Tomkins-Tinch CH, Sochat V, et al. Sustainable data analysis with Snakemake [version 2; peer review: 2 approved]. F1000 research. 2021;10:33-.
101.Huang W, Li L, Myers JR, Marth GT. ART: a next-generation sequencing read simulator. Bioinformatics. 2012;28(4):593-4.
102.Langmead B, Salzberg SL. Fast gapped-read alignment with Bowtie 2. Nat Methods. 2012;9(4):357-9.
103.McMurdie PJ, Holmes S. phyloseq: an R package for reproducible interactive analysis and graphics of microbiome census data. PLoS One. 2013;8(4):e61217-e.
104.Wood DE, Salzberg SL. Kraken: ultrafast metagenomic sequence classification using exact alignments. Genome Biol. 2014;15(3):R46-R.
105.Ondov BD, Bergman NH, Phillippy AM. Interactive metagenomic visualization in a Web browser. BMC Bioinformatics. 2011;12(1):385-.
106.Altschul SF, Gish W, Miller W, Meyers EW, Lipman DJ. Basic Local Alignment Search Tool. Journal of molecular biology. 1990;215(3):403-10.
107.Buchfink B, Xie C, Huson DH. Fast and sensitive protein alignment using DIAMOND. Nat Methods. 2015;12(1):59-60.
108.Menzel P, Ng KL, Krogh A. Fast and sensitive taxonomic classification for metagenomics with Kaiju. Nat Commun. 2016;7(1):11257-.
109.Jiang H, Lei R, Ding S-W, Zhu S. Skewer: a fast and accurate adapter trimmer for next-generation sequencing paired-end reads. BMC Bioinformatics. 2014;15(1):182-.
110.Di Tommaso P, Chatzou M, Floden EW, Barja PP, Palumbo E, Notredame C. Nextflow enables reproducible computational workflows. Nat Biotechnol. 2017;35(4):316-9.
111.Wright RJ, Comeau AM, Langille MGI. From defaults to databases: parameter and database choice dramatically impact the performance of metagenomic taxonomic classification tools. Microb Genom. 2023;9(3).
112.Marcelino V, Holmes EC, Sorrell TC. The use of taxon-specific reference databases compromises metagenomic classification. BMC Genomics. 2020;21(1):184-.
113.Szóstak N, Handschuh L, Samelak-Czajka A, Tomela K, Schmidt M, Pruss Ł, et al. Host Factors Associated with Gut Mycobiome Structure. mSystems. 2023;8(2):e0098622-e.
114.Chen B-Y, Lin W-Z, Li Y-L, Bi C, Du L-J, Liu Y, et al. Characteristics and Correlations of the Oral and Gut Fungal Microbiome with Hypertension. Microbiol Spectr. 2023;11(1):e0195622-e.
115.Blanco-Míguez A, Beghini F, Cumbo F, McIver LJ, Thompson KN, Zolfo M, et al. Extending and improving metagenomic taxonomic profiling with uncharacterized species using MetaPhlAn 4. Nat Biotechnol. 2023.
116.Usyk M, Peters BA, Karthikeyan S, McDonald D, Sollecito CC, Vazquez-Baeza Y, et al. Comprehensive evaluation of shotgun metagenomics, amplicon sequencing, and harmonization of these platforms for epidemiological studies. Cell Rep Methods. 2023;3(1):100391-.
117.Thielemann N, Herz M, Kurzai O, Martin R. Analyzing the human gut mycobiome – A short guide for beginners. Comput Struct Biotechnol J. 2022;20:608-14.

11 APPENDIX

11.1 APPENDIX I

Genome	Species	RefSeq ID
Fungi	*Candida albicans SC5314*	GCF_000182965.3
	Aspergillus lentulus	GCF_010724455.1
	Penicillium rubens Wisconsin 54-1255	GCF_000226395.1
	Saccharomyces cerevisiae S288C	GCF_000146045.2
	Malassezia globosa CBS 7966	GCF_000181695.1
	Debaryomyces hansenii	GCF_000006445.2
	Candida parapsilosis	GCF_000182765.1
	Pichia kudriavzevii	GCF_003054445.1
	Nakaseomyces glabratus	GCF_000002545.3
	Encephalitozoon cuniculi GB-M1	GCF_000091225.2
	Malassezia restricta	GCF_003290485.1
	Trichosporon asahii	GCF_000293215.1
	Candida tropicalis	GCF_000006335.3
	Candida orthopsilosis Co 90-125	GCF_000315875.1
	Candida auris	GCF_002775015.1
	Alternaria burnsii	GCF_013036055.1
	Alternaria ethzedia	GCF_023757985.1
	Candida haemuloni	GCF_002926055.2
	Candida subhashii	GCF_019202705.1
	Acaromyces ingoldii	GCF_003144295.1
	Arthroderma uncinatum	GCF_011692745.1
	Ascochyta rabiei	GCF_004011695.1
	Ascoidea rubescens DSM 1968	GCF_001661345.1
	Alternaria postmessia	GCF_024291825.1
	Alternaria rosae	GCF_020736505.1
	Alternaria triticimaculans	GCF_023758025.1
	Beauveria bassiana ARSEF 2860	GCF_000280675.1
	Bipolaris sorokiniana ND90Pr	GCF_000338995.1
	Blastomyces dermatitidis ER-3	GCF_000003525.1
	Blastomyces gilchristii SLH14081	GCF_000003855.2
	Boeremia exigua	GCF_020726555.1
	Brettanomyces bruxellensis	GCF_011074885.1
	Botrytis porri	GCF_014898465.1
	Chaetomium globosum CBS 148.51	GCF_000143365.1
	Cladophialophora bantiana CBS 173.52	GCF_000835475.1
	Clavispora lusitaniae ATCC 42720	GCF_000003835.1
	Daldinia caldariorum	GCF_022478825.1
	Endocarpon pusillum Z07020	GCF_000464535.1
	Filobasidium floriforme	GCF_021052385.1
	Fonsecaea nubica	GCF_001646965.1
	Gamsiella multidivaricata	GCF_025024155.1
	Halteromyces radiatus	GCF_025201355.1
	Kluyveromyces lactis	GCF_000002515.2
	Kuraishia capsulata CBS 1993	GCF_000576695.1
	Pneumocystis murina B123	GCF_000349005.2
	Pseudozyma hubeiensis SY62	GCF_000403515.1
	Thyridium curvatum	GCF_004353045.1
	Verticillium alfalfae VaMs.102	GCF_000150825.1
	Westerdykella ornata	GCF_010094085.1
	Aspergillus aculeatinus CBS 121060	GCF_003184765.1

11.2 Appendix II

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-
"""
Created on Mon Sep  5 13:13:39 2022

@author: ekateria
"""

from Bio import SeqIO
from Bio.SeqRecord import SeqRecord
from Bio.Seq import Seq
import os
import pandas as pd
import numpy as np

workdir='/cluster/projects/nn9383k/arfa/mockcommunity/MC_genomes'
folders=[x[0] for x in os.walk(workdir)]

for f in folders[1::]:
    files=os.listdir(f)
    newdir=f.replace('refseq','concatenated')
    os.mkdir(newdir)
    for fi in files[1:]:
        qfasta=SeqIO.parse('/'.join([f, fi]),'fasta')
        concat=str('')
        for fasta in qfasta:
            concat=''.join([concat,str(fasta.seq)])
        record = SeqRecord(Seq(concat, name=fi.replace('.fna',''),
                           id=fi.replace('.fna',''), description=''))
        SeqIO.write(record,'/'.join([newdir,fi]),'fasta')

##Concatenate all files into 'fungi.fna' using cat *.fna > fungi.fna
## in terminal


## Find lengths of each genome
workdir='/cluster/projects/nn9383k/arfa/mockcommunity/MC_genomes'
summary=pd.read_csv('/'.join([workdir, 'Fungal_MC_GCF.txt']), sep='\t')
fdir='/'.join([workdir, 'concatenated/all'])
files=os.listdir(fdir)
for fi in files[1:]:
    qfasta=SeqIO.parse('/'.join([fdir,fi]),'fasta')
    for fasta in qfasta:
        x='_'.join(fasta.name.split('_')[:2])
        summary.loc[summary['RefSeqID']==x,'Length, bp']=len(fasta)

summary.to_csv('/'.join([workdir, 'Fungal_MC_GCF.txt']),index=False)
```

```
##Read the simulated dataset (check 100000 reads per sequence)
workdir='/cluster/projects/nn9383k/arfa/mockcommunity/MC_genomes'
qfastq=SeqIO.parse('/'.join([workdir,'MC_100000_1.fq']),'fastq')
fastq_sum=pd.DataFrame(columns=['ReadName','AvgQual'])
for seq in qfastq:
    fastq_sum = pd.concat([fastq_sum,
pd.DataFrame([['_'.join(seq.name.split('_')[:2]),
                      np.mean(seq.letter_annotations['phred_quality'])]],
                      columns=['ReadName', 'AvgQual'])], ignore_index=True)

#Check how many reads per genome there are
reads_per_seq=fastq_sum['ReadName'].value_counts()

#Check the average quality of the reads per each genome
qual_per_genome=pd.DataFrame(fastq_sum.groupby('ReadName')['AvgQual'].mean(),
columns=['AvgQual'])
qual_per_genome=qual_per_genome.reset_index()
qual_per_genome=qual_per_genome.rename(columns=(99))
summary=summary.merge(qual_per_genome, on='RefSeqID', how='left')
summary=summary.drop(columns='AvgQual_x')
summary=summary.rename(columns={'AvgQual_y':'AvgQual'})
summary['NumReads']=100000
summary.to_csv('/'.join([workdir, 'Fungal_MC_GCF.txt']),index=False, sep
='\t')
```

11.3 Appendix III

```
"""
Created on Mon 31 Oct 2022 10:26:40 AM CEST

Author: Arfa Irej Qureshi
Title: Merging MMC read 1 and read 2
"""
#Set working directory
setwd("/ess/p1068/data/durable/007-f_smei/001-
trro/CRCbiome/development/Arfa/MetaPhlAn/")

#Read metaphlan_read1.txt
read1 = read.delim("metaphlan_read1.txt", header = TRUE, sep = "\t", quote =
"", dec = ".")

#Read metaphlan_read2.txt
read2 = read.delim("metaphlan_read2.txt", header = TRUE, sep = "\t", quote =
"", dec = ".")

#Merge both files
read_both = merge(read1, read2, by.x = 1, by.y = 1, all.x = FALSE)

#Check if the files were merged correctly
head(read_both)

#Write back the file
write.table(read_both, file = "read_both.txt", col.names = TRUE, row.names =
FALSE, quote = FALSE, sep = "\t")
```

11.4 Appendix IV

```
"""
Created on Thu 24 Nov 2022 03:30:05 PM CEST

Author: Arfa Irej Qureshi
Title: Converting MetaPhlAn output to PhyloSeq object
"""

#Set working directory
setwd("/ess/p1068/data/durable/007-f_smei/001-
trro/CRCbiome/development/Arfa/MetaPhlAn")

#Read table
taxa <- read.table("read_both.txt", sep = "\t", header = TRUE)

#Run tidyverse
library(tidyverse)

#Remove NCBI tax id and additional species columns
taxa %>%
select(- NCBI_tax_id.x, - NCBI_tax_id.y, - additional_species.x, -
additional_species.y) -> taxa

#Dividing names into unique columns
taxa %>%
separate(clade_name, sep = "\\|", remove = FALSE, into = c("Kingdom",
"Phylum", "Class", "Order", "Family", "Genus", "Species")) %>%
select(-clade_name) -> taxa

#Filtering for k__Eukaryota
taxa %>%
filter(Kingdom=="k__Eukaryota")->taxa

#Removing relative abundance from taxa table
taxa %>%
select(1:7) -> tax_mat

#Naming OTU rows
rownames(tax_mat) <- paste0("OTU", 1:nrow(tax_mat))

#Must have taxa table in as.matrix format for PhyloSeq
tax_mat <- as.matrix(tax_mat)

#Create OTU table
OTU <- taxa[,8]
OTU <- round(OTU)
OTU <- as.data.frame(OTU)
rownames(OTU) <- paste0("OTU", 1:nrow(OTU))
```

```
OTU <- as.matrix(OTU)

#Run Phyloseq
library(phyloseq)

#Transforming to a PhyloSeq object
OTU <- otu_table(OTU, taxa_are_rows = TRUE)
TAX <- tax_table(tax_mat)

#Create Phyloseq object
read_both <- phyloseq(OTU, TAX)

#Create bar graph from Phyloseq object
plot_bar(read_both, fill = "Species")
```

11.5 Appendix V

```
"""
Created on Tue 14 Feb 2023 02:19:23 PM CEST

Author: Arfa Irej Qureshi
Title: Custom HMS database
"""

#!/usr/bin/bash
# Job name:
#SBATCH --job-name=HMSPipeline

# Project:
#SBATCH --account=nn9383k

# Wall time limit:
#SBATCH --time=00-18:00:00

# Memory:
#SBATCH --nodes=6
#SBATCH --ntasks-per-node=1
#SBATCH --cpus-per-task=16
#SBATCH --mem-per-cpu=6G

## Set up job environment:
set -o errexit # Exit the script on any error
set -o nounset # Treat any unset variables as an error

## Enable same name autoswapping
LMOD_DISABLE_SAME_NAME_AUTOSWAP=no

##Load modules:
module --quiet purge
module load Bowtie2/2.4.2-GCC-10.2.0
module load SAMtools/1.11-GCC-10.2.0
module load BLAST+/2.11.0-gompi-2020b
module load R/4.0.3-fosscuda-2020b

##Make custom database:
bash /cluster/projects/nn9383k/arfa/hms_analysis/custom_db_creation_large.sh -
d Custom_DB.fasta -l 500 -m /cluster/projects/nn9383k/arfa/hms_analysis/
```

11.6 Appendix VI

```
"""
Created on Sun 30 Apr 2023 09:28:58 PM CEST

Author: Arfa Irej Qureshi
Title: Mapping FMC to FindFungi's 32 Kraken DB
"""

DBrange=list(range(1,33))
rule all:
    input:
        expand("/cluster/projects/nn9383k/arfa/kraken2_analysis/FindFungi_32_K
raken/Kraken_{id}/UNCLASSIFIED_{id}#.fq", id=DBrange),
        expand("/cluster/projects/nn9383k/arfa/kraken2_analysis/FindFungi_32_K
raken/Kraken_{id}/CLASSIFIED_{id}#.fq", id=DBrange),
        expand("/cluster/projects/nn9383k/arfa/kraken2_analysis/FindFungi_32_K
raken/Kraken_{id}/output_{id}.txt", id=DBrange),
        expand("/cluster/projects/nn9383k/arfa/kraken2_analysis/FindFungi_32_K
raken/Kraken_{id}/report_{id}.txt", id=DBrange),
        expand("/cluster/projects/nn9383k/arfa/kraken2_analysis/FindFungi_32_K
raken/combined_classified.fq")
rule Kraken2:
    output:
        output1="/cluster/projects/nn9383k/arfa/kraken2_analysis/FindFungi_32_
Kraken/Kraken_{id}/UNCLASSIFIED_{id}#.fq",
        output2="/cluster/projects/nn9383k/arfa/kraken2_analysis/FindFungi_32_
Kraken/Kraken_{id}/CLASSIFIED_{id}#.fq",
        output3="/cluster/projects/nn9383k/arfa/kraken2_analysis/FindFungi_32_
Kraken/Kraken_{id}/output_{id}.txt"
    threads:
        16
    log:
        "logs/kraken{id}.log"
    shell: '''
    kraken --db
/cluster/projects/nn9383k/arfa/findfungi_analysis/KrakenDB/Kraken_32DB/Kraken_
{wildcards.id}/ --fastq-input --threads {threads} --unclassified-out
{output.output1} \
    --classified-out {output.output2} --output {output.output3} \
    --paired
/cluster/projects/nn9383k/arfa/mockcommunity/fungalmockcommunity/mock_fungal_p
aired1.fq \
    /cluster/projects/nn9383k/arfa/mockcommunity/fungalmockcommunity/mock_fung
al_paired2.fq \
    &> {log}
    '''
localrules: krakenreport
rule krakenreport:
```

```
    input:
        expand("/cluster/projects/nn9383k/arfa/kraken2_analysis/FindFungi_32_K
raken/Kraken_{{id}}/output_{{id}}.txt", id=DBrange)
    output:
        output="/cluster/projects/nn9383k/arfa/kraken2_analysis/FindFungi_32_K
raken/Kraken_{id}/report_{id}.txt"
    shell: '''
        kraken-report --db
/cluster/projects/nn9383k/arfa/findfungi_analysis/KrakenDB/Kraken_32DB/Kraken_
{wildcards.id}/ \
        /cluster/projects/nn9383k/arfa/kraken2_analysis/FindFungi_32_Kraken/Kr
aken_{wildcards.id}/output_{wildcards.id}.txt \
        > {output.output}
    '''
```