Norwegian University
of Life Sciences

Master's Thesis 2022 30 ECTS
Faculty of Chemistry, Biotechnology and Food Science (KBM)

Reduced Metagenomic Sequencing Resolution of the Human Gut Using HumGut as a Reference Database

Helene Drennan Olsen
Chemistry and Biotechnology (M.Sc.) - Bioinformatics

Summary

The gut microbiome is a source of great genetic diversity in humans and is prevalent in topics concerning human health and diseases. Today, most compositions of human gut microbiomes are estimated on 16S amplicon sequencing and shotgun sequencing data. Although 16S amplicon sequencing is cheap and fast, it does not gain a taxonomic resolution down to species level like the more costly and computationally expensive shotgun sequencing. Reduced Metagenomic Sequencing (RMS) has been suggested as an alternative method, as it has been shown to gain a taxonomic resolution down to species level and is more cost-effective per sample than shotgun sequencing. However, it requires a good reference database. The HumGut genome collection was used as a reference database in this thesis, and it contains >30 000 genomes prevalent in a healthy human gut. This thesis investigated the taxonomic resolution RMS can achieve using HumGut as a reference database. Further, a dataset from the PreventADALL-study was provided, containing human gut samples from mother-child pairs. The data was used to investigate whether a vertical transmission of *Bifidobacterium* could be detected in the samples.

The HumGut dataset was divided into subgroups based on the genomes' genus, as genome clustering of all genomes prove to be too computationally demanding. The genomes within most genera were able to cluster to a condition value below 10, meaning the lowest possible taxonomic resolution was obtained but with the advantage of more stable abundancy estimates which is important when investigating vertical transmission. Results indicated that the genera containing more RMS fragments per genome returned a higher taxonomic resolution, some even down to strain level, while genera with fewer RMS fragments per genome returned a lower resolution, some displaying a resolution not much lower than genus level. *Bifidobacterium* were of the genera that obtained a lower taxonomic resolution, resulting in a >98% reduction of genomes after genome clustering. As a result, no correlation of *Bifidobacterium* distribution was found between mother and child.

Sammendrag

Tarmflora er en kilde til stort genetisk mangfold i mennesket, og er et svært omtalt emne som angår både menneskehelse og sykdommer. Sammensetningen av mikroorganismer i menneskets tarm er hovedsakelig basert på 16S amplikonsekvensering og shotgun sekvensering. Selv om 16S amplikonsekvensering er billigere og raskere, så får den ikke en taksonomisk oppløsning ned til artsnivå som den mer ressurskrevende og tregere shotgun sekvenseringen. Redusert metagenomisk sekvensering (RMS) har blitt foreslått som en alternativ metode, da den har vist å kunne få en taksonomisk oppløsning ned til artsnivå og er billigere enn shotgunsekvensering. Metoden krever en god referansedatabase. I denne oppgaven ble referansedatabasen HumGut brukt. HumGut består av genomer som eksisterer i en sunn menneskelig tarm, og har over 30 600 oppføringer. Denne oppgaven undersøkte den taksonomiske oppløsningen RMS kan oppnå ved å bruke HumGut som referansedatabase. Videre ble prøver fra tarmen til mor-barn par I et datasett fra PreventADALL-studien undersøkt. Det ble undersøkt om tegn til vertikal overføing av *Bifidobacterium* kunne påvises i prøvene.

HumGut ble delt inn i grupper basert på genomenes slekt, da klynging av genomer viste seg å være for krevende å beregne. Genomene innen de fleste slekter kunne gruppere seg til en tilstandsverdi under 10, som betyr at lavest mulig taksonomisk oppløsning ble oppnådd, men med fordelen av mer stabile estimater av mengder mikroorganismer i prøvene. Dette er viktig når vertikal overføring skal undersøkes. Resultatene indikerte at slektene med flere RMS-fragmenter per genom ga en høyere taksonomisk oppløsning, der noen fikk en oppløsning ned på stammenivå. Slektene med færre RMS-fragmenter per genom ga en lavere taksonomisk oppløsning, der enkelte slekter ikke fikk en oppløsning lavere enn på slektsnivå. *Bifidobacterium* var en av slektene som oppnådde en lavere taksonomisk oppløsning, noe som resulterte i >98% reduksjon av genomer etter klyngning. Resultatene indikerte ikke noen direkte sammenheng mellom *Bifidobacterium*-sammensetningen mellom mor og barn prøvene.

Acknowledgements

This work presented in this thesis was performed at the Faculty of Chemistry, Biotechnology and Food Sciences (KBM), at the Norwegian University of Life Sciences (NMBU), under the supervision of Lars Snipen in 2022.

I would like to thank Lars Snipen, my thesis supervisor, for all the excellent guidance and support throughout this thesis. You are very knowledgeable and patient, and I have learned a lot about my thesis subject and other relevant topics within the microbial genomics field in the weekly seminars you were so kind to invite me to.

To my family and friends: thank you for all the love and support making my 5.5 years at NMBU such an amazing journey! A special thanks to my partner, Håkon, for reminding me to take breaks, always providing support and for making me laugh during the tough days. Also thank you to Trude for being a great sparring partner throughout these past years, I really value the conversations we have had. Lastly, a huge thanks to Madeleine for proof-reading my thesis and constantly rooting for me, your enthusiasm and support have been invaluable to me.

Table of Contents

List of Figures

List of Abbreviations

dd-RADseq	double-digested Restriction Site Associated DNA sequencing
DNA	Deoxyribonucleic Acid
HPC	High-Performance Computing
MAGs	Metagenome-Assembled genomes
NCBI	The National Center for Biotechnology Information
NGS	Next Generation Sequencing
OTU	Operational Taxonomic Unit
PCR	Polymerase Chain Reaction
RMS	Reduced Metagenomic Sequencing
rRNA	Ribosomal Ribonucleic Acid
WGS	Whole Genome Sequencing

1. Introduction

Microbial communities are found everywhere, from the human gut (Hiseni *et al.,* 2021) and soil environments (Handelsman *et al.*, 1998), to extreme environments found in hot springs (Massello *et al.*, 2020) and saline lakes (Naghoni *et al*., 2017). Despite microbe's small size, they impact life on every scale, from small human infections to being a crucial part in cycling elements that are critical for sustaining life on earth (American Society for Microbiology, 2003). The study of microbial community compositions, and shifts within them, is therefore important to understand their role both in humans and our environment. In the past decade, studies of the composition of these communities have increased rapidly along with methods to investigate them (Bretweiser *et al.*, 2019). There are two well established approaches today: amplicon sequencing and shotgun sequencing (Snipen *et al.*, 2021). Studies using methods like these to sequence DNA from a community of microorganisms, fall under the field of metagenomics.

1.1 Metagenomics

An organism's complete set of DNA is called a genome, and the collection of genomes found in an environmental sample is labelled the metagenome of that sample community. The term "metagenome" was first coined by Handelsman *et al.* (1998), when describing the cloning of the collective genome of soil microflora. Metagenomics refers to the study of the structure and function of a metagenome, and often from a specific community of microorganisms from an environment like soil or the human gut. The metagenomes are usually big and intricate, and analyzing them can result in large, complex datasets (Breitiwieser *et al.*, 2019). To be able to analyze metagenomes one needs, amongst other things, a sequencing technique advanced enough to handle such intricate compositions of genetic material.

The introduction of Next-Generation Sequencing (NGS) in the 2000s (Reinartz *et al.*, 2002), facilitated the growth and development of the field of metagenomics. NGS was preceded by Sanger sequencing, which had been the leading sequencing technique since it was introduced in the late

1970s (Sanger, 1977). In Sanger sequencing, isolated DNA strains are sequenced by a terminal chain reaction. The technique sequences one DNA fragment at a time, making sequencing of entire genomes, let alone metagenomes, an expensive and time-consuming task. Furthermore, the method has been dependent on cultivating a pure single-strain bacteria culture to isolate their DNA before sequencing (Cermak *et al.*, 2020). A major issue is that only a small minority of microorganisms can be cultured in the laboratory (Stewart, 2012), thus, making it hard to discover the likely composition of a metagenome using Sanger.

With the development of NGS, came sequencing techniques that did not demand the need of culturing microorganisms prior to sequencing them. Wooley & Ye, 2009, even defined metagenomics as the study of microbial communities sampled directly from their natural environment *without* prior culturing. Furthermore, NGS technology made it possible to sequence many reactions in parallel on a microscopic scale, leading to sequencing becoming faster, more cost effective, and less labor intensive than Sanger (Goodwin *et al*., 2016). In example, the sequencing of a human genome was estimated to cost 100 million USD in 2001. Further development of NGS technology has reduced this cost to under 1000 USD in 2021 (National Human Research Institute, 2021). The faster and cheaper NGS techniques, with its massive parallel sequencing, also made it easier to sequence genomes with a much larger read-depth. This is a major advantage when analyzing complex compositions like metagenomes, where a higher sequencing depth allows for rarer and less abundant genomes to be detected. Today, both amplicon and shotgun sequencing are popular techniques to combine with different NGS, like Illumina, to sequence metagenomes.

Amplicon sequencing, also known as metabarcoding when applied to metagenomic studies, can be applied to reveal the taxonomic composition in environmental samples. The technique targets one or multiple marker genes via specific primers and amplifies the genomic area through polymerase chain reactions (PCR). The amplified marker genes, dubbed amplicons, are then sequenced using a NGS technique, like the Illumina sequencing platform. Typically, the 16S rRNA gene is used as a phylogenetic marker gene in amplicon sequencing of microorganisms. This is due to the gene being present and highly conserved in almost all prokaryotes, however, it also

contains hypervariable regions that demonstrate sequence diversity among different bacteria (Schmalenberger *et al.*, 2001). The hypervariable regions are flanked by regions that are highly conserved across prokaryotes, allowing the use of universal primers to amplify 16S rRNA across a large fraction of prokaryotes (Baker *et al*., 2003). By sequencing these variable elements in microorganisms from an environmental sample, taxonomic composition and estimated relative abundances of the sample can be found. On the other hand, using a highly conserved gene like the 16S rRNA reduces the methods taxonomic resolution (Snipen *et al.*, 2021). This leads to the method having difficulties in separating genomes in environmental samples down to species and strain resolution.

An alternative method with higher resolution is shotgun sequencing, also known as whole genome shotgun (WGS) sequencing. Shotgun sequencing, in broad strokes, involves fragmenting the DNA into many small pieces at random, sequencing them, then stitching it back together using bioinformatic pipelines. This means unlike amplicon sequencing; shotgun sequencing sequences all genomic material in a sample directly with an NGS technique. However, analyzing and stitching back together fragments from an entire metagenome leads to a computationally heavier and more costly sequencing compared to 16S amplicon sequencing (Sims *et al*., 2014; Snipen *et al*., 2021). In return, the taxonomic resolution is higher than with 16S sequencing. Using shotgun sequencing, taxonomic resolution down to species and even strain level can be found if the species and strains are different enough, and if the samples are sequenced deep enough (Durazzi *et al.*, 2021; Snipen *et al.*, 2021).

Regarding the taxonomic resolution, it should be mentioned that the microbial world is more diverse and complex than predicted by scientists who made the taxonomic system centuries ago (Bretweiser *et al*., 2017). This leads to possible ambiguities and struggles when classifying some microorganisms after the taxonomic ranking system. The ambiguities, combined with the continual development of modern technologies promoting new discoveries, can lead to rapid changes in taxonomy for certain microorganisms.

When sequencing metagenomes and deciding on a sequencing method, this trade-off of the cheaper 16S amplicon sequencing and the more expensive but higher taxonomic resolution shotgun sequencing must be considered. The problem is that in larger studies where a high taxonomic resolution is important, shotgun sequencing can be considered too expensive and time consuming (Ravi *et al.*, 2018). However, the 16S amplicon sequencing might not be a suitable option either due to the lower taxonomic resolution. Therefore, there is motivation to find a method that is cheap and relatively fast, like amplicon sequencing, but with a higher taxonomic resolution on par with shotgun sequencing. A method named Reduced Metagenomic Sequencing (RMS) has been suggested to fill such a position (Liu *et al.*, 2017; Ravi *et al.*, 2018; Snipen *et al.*, 2021).

1.1.2 Reduced Metagenomic Sequencing

Reduced Metagenomic Sequencing (RMS) is based on double-digested Restriction Site Associated DNA sequencing, abbreviated dd-RADseq, combined with Illumina sequencing (Liu *et al.*, 2017; Snipen *et al.*, 2021). The method uses sequence specific endonucleases to fragment genomic DNA by restriction digestion, thereby reducing the genome sequence space and genome complexity (Hess *et al.*, 2020; Snipen *et al.*, 2021). It uses two different restriction endonucleases simultaneously, which is referred to as double restriction digestion. The constructed fragments are flanked by each targeted restriction site. Following the fragmentation, the fragments are amplified by PCR before they are sequenced. It should be noted that the variable lengths and compositions of the fragments can lead to variable PCR-amplicon efficiency and biases (Snipen *et al.*, 2021).

RMS has similarities with both shotgun and amplicon sequencing. It creates many different fragments with variable number and size between genomes, like shotgun sequencing. However, a genome will also produce the exact same fragments and reads each time it is copied, as seen in amplicon sequencing. Snipen *et al.* 2021 showed that RMS can be used to profile microbial communities down to species level and even strains in some cases. Strains deemed identical after 16S sequencing, were clearly discriminated by RMS since the genomes would differ in number of RMS fragments by quite a bit. Furthermore, RMS reduces the sequencing efforts compared to shotgun sequencing, which also reduces the cost-per-sample (Snipen *et al.*, 2021). Thus, RMS is

faster and cheaper than shotgun sequencing, but may return a higher taxonomic resolution compared to 16S amplicon sequencing.

There can arise difficulties when using the RMS approach to infer taxonomic compositions of reads. Just as the amplicons in amplicon sequencing can be clustered to represent some taxon, the fragments in RMS may also be clustered. However, each taxa produces variable numbers of distinct fragments, making it difficult to infer a taxonomic composition from the clusters alone (Snipen *et al.*, 2021). The RMS approach, like the shotgun sequencing approach, requires a reference database to map the cluster sequences to create taxonomic profiles.

1.2 HumGut

The gut microbiome is a source of great genetic diversity in humans, and it is estimated that the metagenome of the human gut contains at least 100 times more genes than the human genome (Gill *et al*., 2006). Metagenomic analysis of the gut microbiome can lead to more comprehensive examinations into how the gut microbiome is colonized, and how it evolved and contributes to both health and disease in humans in response to the environment over time. A major issue in studies of the gut microbiome has been the lack of a comprehensive genome collection to be used as a reference database when sequencing samples from human gut (Hiseni *et al.*, 2021), and the HumGut project sought to rectify this.

The HumGut project aimed to collect the most prevalent prokaryotic genomes found in a healthy human gut, in order to function as a reference database (Hiseni *et al.*, 2021). It includes mostly Metagenomic-assembled genomes (MAGs) from human gut, and some RefSeq genomes (O'Leary *et al.*, 2016). MAGs are microbial genomes that are reconstructed, *de novo*, from metagenomic data. There are risks when reconstructing a genome this way, as contigs contributing to a MAG could derive from a different genome. This could lead to erroneous areas in MAGs, which again could lead to variable qualities in genomes constructed this way.

Over 5 700 healthy human gut metagenomes were screened to see if they contained any of the > 490 000 publicly available prokaryotic genomes sourced from RefSeq and the UHGG collection (Hiseni *et al.*, 2021). The resulting genomes were scored and ranked by prevalence in the healthy human metagenomes. Genomes were clustered to a 97.5% sequence identity resolution, and the HumGut collection is comprised of these clustered genomes. Lastly, the HumGut collection was found to outperform both standard Kraken2 database and the UHGG collection making it a contender as a reference database for human gut samples.

Recent investigations have indicated that a difference in strain level in human gut may be crucial for phenotypic differences (Snipen *et al*., 2021). If a study wants to capture phenotypic differences, many samples are often required to capture the biological variation. In order to capture the resolution down to strain level, full shotgun sequencing is necessary as the 16S amplicon sequencing lacks the higher taxonomic resolution. The cost of using full shotgun sequencing may limit the studies requiring many samples, so there is an interest to see if it is possible to use RMS to gain the higher taxonomic resolution in a cheaper and faster way.

1.2.1 The *Bifidobacterium* Genus

Bacteria from the *Bifidobacterium* genus were first discovered in the early 1900s by the French pediatrician Henry Tissier while working on stools from breast-fed infants (Tissier, 1900; Killingstad, 2021). The bacteria are recognized as GC-rich, gram-positive, anaerobic, non-motile, non-spore-forming, polymorphic rods (Wong et al., 2020). This genus is known to have a presence in the human gut microbiome in both adults and children. Therefore, it is not surprising that the HumGut genome collection contains *Bifidobacterium*-genomes. There are 328 *Bifidobacterium* genomes present in HumGut, meaning about 1% of the collection comprises of *Bifidobacterium* genomes.

Bifidobacterium is considered important in human gut health due to their role in gut development and defense system (Makino *et al*., 2013), therefore the genus has been the topic of several studies. The bacterial genus has been shown to be one of the most abundant in healthy infant guts (Odamaki

et al., 2016), and how the bacteria colonize the gut is still a discussed topic. Furthermore, strains of *Bifidobacterium* have been found in breastmilk in mammals, suggesting it is a possible transmission route from mother to child (Laursen *et al.,* 2021). The abundance of *Bifidobacterium* in infant gut is shown to decrease after weaning and continues to decrease with age (Odamaki et al., 2016).

1.3 Aim of Study

Today, most compositions of microbial communities are estimated based on 16S amplicon sequencing data or shotgun sequencing metagenomic data. There is a trade-off between the two methods as amplicon sequencing is cheaper and faster but gains a lower taxonomic resolution than the more costly and computationally expensive shotgun sequencing. Reduced Metagenomic Sequencing (RMS) has been suggested, as a sort of compromise between the two other methods. RMS is more cost-effective than shotgun sequencing and has shown a higher taxonomic resolution than 16S amplicon sequencing down to species level, however it requires a good reference database (Snipen *et al.,* 2021).

Research on the composition of the human gut usually calls for a higher taxonomic resolution than what is gained with 16S amplicon sequencing. However, the more costly shotgun sequencing could be quite limiting in large scale studies. There is an interest to see if RMS could be used for these research purposes. This thesis aims to answer,

- To what taxonomic resolution can RMS gain on samples from the human gut using HumGut as a reference database?

In addition to the taxonomic resolution, there was also an interest in mapping reads and estimating abundances for a genus' genome-clusters in human gut samples using this approach. To investigate this, a dataset containing 16 mother-child samples from the PreventADALL-study was provided to this thesis (Carlsen *et al.,* 2018; Nilsen 2022). *Bifidobacterium* was chosen as a genus, as we

know it appears in both mother and child gut microbiomes. The aim with the provided dataset was to see,

- Is there a correlation between the distribution of *Bifidobacterium* between mother and child?

To detect this, a sufficient taxonomic resolution and good estimates of abundances are most likely required.

2. Methods

The data analysis and wrangling were carried out using RStudio 4.0.4 (R Core Team, 2021) on NMBU's high performance computer (HPC) called Orion. The steps using the microRMS R package followed "Tutorial 1 – the microbial community composition" found in the package's Readme.md file (Snipen, 2021). All visualizations were made using the *ggplot2* package in R (Wickham, 2016). The VSEARCH tool was used for clustering genomes and RMS fragments, and for mapping reads (Rognes *et al.*, 2016). The most essential R and shell scripts can be found in the Appendix.

2.1 The HumGut Data

HumGut was chosen as a reference database in this thesis to investigate the resolution the RMS method has on samples from the human gut (Hiseni *et al.*, 2021). The HumGut collection was loaded into R as a data frame with one row per genome and 24 different genome-features as columns. As of 03.10.2022, there are 30 614 rows in the data frame, and thus 30 614 genomes in HumGut. Not all 24 genome features were needed, so the data frame was filtered to include the 9 features shown in table 1.

Table 1: An overview of the selected genome features in the HumGut data frame, and the genome information they contain.

Column name	**Column information**
genome_id	The genome's unique HumGut ID
genome_size	The genome size in basepairs (L)
GC	The genome's GC content
genome_type	Assembly type of the genome (MAG, Complete Genome, Contig etc.)
Source	Whether the genome is from UHGG or RefSeq
ncbi_organism_name	The organisms name according to the NCBI database *
ncbi_tax_id	The organisms unique NCBI tax ID
ncbi_rank	The lowest level of taxonomic hierarchy of the genome**

path	Path to the genome's fasta file directory
genome_file	The genome's filename. The file is a fna file which is gz compressed. The filename is named after the genome's HumGut ID (genome_id.fna.gz)

* (Schoch *et al.*, 2020; Sayer *et al.*, 2019)
** Strain, serotype, subspecies and "no rank" are included as ranks here, although they are not official taxonomic ranks.

2.2 VSEARCH

VSEARCH is a 64-bit tool that is used for clustering and processing metagenomic data in this thesis. According to the article by Rognes *et al.* 2016, the tool was designed as an alternative to the widely used USEARCH tool (Edgar, 2010). Both tools contain most of the same functions, however, VSEARCH is open-sourced and has made their 64-bit version free of charge. Furthermore, VSEARCH has shown results both better than and on par with results from USEARCH (Rognes *et al.*, 2016). Consequently, VSEARCH has been utilized in several of the *microrms*-package functions (Snipen, 2021).

2.2.1 Clustering

VSEARCH performs *de novo* clustering of sequences using a greedy and heuristic centroid-based algorithm with an adjustable sequence similarity threshold (Rognes *et al.*, 2016). In short, a greedy and heuristic algorithm focuses on local optimal solutions, rather than a general optimal global solution, gaining the advantage of faster computational time. *De novo* clustering refers to clustering the sequences based on the similarity to the other input sequences, and not to a reference database. Consequently, the computational cost of a *de novo* clustering scales quadratically with the number of unique sequences. Further, the input sequences can be processed in different ways before clustering, and in this thesis all clustering was done with the *cluster_fast* option that pre-sorts the sequences based on sequence length.

In clustering, the input sequence is used as a query in a search against a database of centroid sequences (Rognes *et al.*, 2016). As this is *de novo* clustering, the database is initially empty of centroid sequences – so the first sequence is set as the centroid in a cluster and added to the

database. Further query sequences are then clustered with the first centroid that shows a similarity equal to or above the id threshold, hence the algorithm is heuristic. If the query sequence does not match to any centroids, it becomes the centroid of a new cluster which is then added to the database. In VSEARCH there is an option to use multi-threaded clustering, meaning several query sequences are searched against the database in parallel. If there are two or more query sequences that do not find a match in the database, they are compared to each other before giving rise to new centroids. Multi-threading is used in this thesis to speed up the clustering process.

2.2.2 Global Pairwise Alignment

VSEARCH can do a global pairwise sequence comparison which can be used to map reads to a database of sequences. The function is called *usearch_global*, and it performs an initial heuristic filtering using shared *k*-mers, and then makes an optimal alignment of the query and the most promising candidate from the database – much like the clustering function (Rognes et al., 2016). Comparing *k*-mers is a faster method to assess the similarity between two sequences, rather than the more time-consuming job of aligning them. The *k*-mers consist of *k* consecutive nucleotides of a sequence, which is set as 8 by default in VSEARCH. That means a sequence of length *n* contains $n - k + 1$ unique *k*-mers at most, including overlapping *k*-mers. VSEARCH counts the *k*-mers that match between the query sequence and the database sequences, counting each *k*-mer that matches only once. The database sequence with the largest number of matching *k*-mers to the query sequence is considered first, and if the alignment indicates a similarity equal to or greater than the id-threshold, the query is mapped to the database sequence. Should several database sequences have the same amount of matching *k*-mers, the shortest sequence is considered first. If the query sequence does not match or align to a database sequence over the id-threshold, the query sequence is rejected.

2.3 Making the RMS-fragments *in silico*

In this section, the RMS-fragments for each HumGut genome were made *in silico* before they were clustered using VSEARCH and a 99% similarity threshold. This resulted in an RMS object for the HumGut genomes.

The job of making RMS-fragments for all genomes was divided into several SLURM array-jobs on Orion to save computational resources, especially time. A shell script was made to run 307 array jobs, each calling on the same RScript to mill through and make RMS fragments for a specific set of genomes. All jobs, except the last one, iterated through a set of 100 genomes each. The last array job iterated through the last 14 genomes in HumGut. Each array job used its SLURM array task ID as input to the RScript, instructing which row in HumGut to be the first in its iteration set. The output of each RScript was a set of new fasta files, one per genome, containing the genomes' RMS fragments.

Every iteration handled one genome at a time in the RScript's designated set of genomes. Each iteration started with reading the genome's fasta file into R by using the *readFasta()*-function from the package *microseq* (Liland *et al.*, 2021). Then, the function *GetRMSfragments()* from the *microrms*-package was used to make the genomes RMS fragments (Snipen, 2021). The function had the genome's fasta file and unique HumGut ID as input. Further, the default settings for the parameters left, right, max and min were used. This meant that the RMS fragments were determined by the default restriction enzymes EcoRI (left) and MseI (right), with the cutting motifs GATTC and TTAA, respectively. The fragments were also determined to be between 30 (min) and 500 base pairs (max), as fragment sizes smaller than 30 or larger than 500 tend to introduce length biases in the data (Snipen *et al*., 2021). The function returns a table where each row is an RMS fragment along with two columns; one containing its header, with fragment information like the fragment's unique fragment ID, and the other containing the sequence, which is the actual RMS fragment. Lastly, the returned table was written into fasta format by using the *writeFasta()* function from *microseq*. The function saved the information as a gz compressed .fna file in a specified directory, maintaining the genome's unique HumGut ID as its filename.

Each genome's expected number of RMS fragments and total number of fragments were added as two columns in the HumGut data frame. The expected number of fragments, had this been random DNA, was calculated using the genome's length and GC-content. The GC-content has to be

included since the restriction enzymes used in this thesis have GC-poor motifs, meaning the genome's GC-content will influence its number of RMS fragments. The longest of the motifs for the two restriction enzymes will be the least likely to occur in the genome sequence, and therefore the limiting factor in determining RMS-fragments. In this thesis the longest motif is EcoRI's GATTC, which is used in formula 1 to determine the expected number of RMS-fragments for a genome.

(1) $$E(n) = \left(\frac{n_{GC}}{2}\right)^2 \times \left(\frac{(1-n_{GC})}{2}\right)^4 \times n_L$$

Where,
$E(n)$ = the expected number of RMS-fragments in a genome,
n_{GC} = the GC-content in genome n,
n_L = length of genome n in bp.

The expected number of RMS-fragments was calculated for each genome and added as a column to the HumGut data frame. Further, as each RMS fragment is one row in the genome's RMS fragment fasta file, the number of rows in the fasta file was set as the total number of RMS fragments for each genome.

Now each genome has a file with its RMS-fragments, however, fragments that are terribly similar will be hard to differentiate between when sequencing using RMS. So, if a genome contains several similar fragments, or a fragment is similar between several genomes, it will be hard to identify where the fragment originated from when sequencing. Clustering these fragments based on similarity will give an insight into how many fragments are quite similar and which genomes they stem from.

2.3.1 Making RMS Objects from Each Genome's RMS Fragments

Several RMS objects were made in this thesis by using the *RMSobject()* function from the *microrms* package (Snipen, 2021), which in turn uses VSEARCH to cluster fragments by similarity. The function takes in RMS fragment fasta files of a selected set of genomes, along with

necessary genome information, and clusters its fragments. This results in an R object which is a list containing a sparse copy number (cpn) matrix and two tables with cluster (cluster.tbl) and genome (genome.tbl) information.

The RMSobject() function has 9 input arguments. Two of the arguments are genome.tbl, a table with metadata of the selected set of genomes, and frg.dir, the path to the directory with the RMS fragment fasta files. Another input argument is a string containing the VSEARCH executable command and, in this case, it was run as a singularity container on Orion. Additionally, the thread argument was set to 10 as VSEARCH allows multi-threading. The default settings were used for the input arguments, identity, min.length, max.length, verbose and tmp.dir. This means the sequence identity for clustering fragments was set to 0.99 (99%), and only fragments of lengths between 30 (min) to 500 (max) base pairs were considered. The function also required a temporary directory, denoted tmp.dir, to store temporary output which is deleted towards the end.

The function returns an object containing the tibbles Cluster.tbl and Genome.tbl, and the sparse matrix Cpn.mat. The Cluster.tbl contains data about all fragment clusters, where each row represents a cluster. Information like the unique cluster name (Cluster), and how many (N.genomes) and which (Members) genomes contain each fragment are included in the table. Genome.tbl is a copy of the input genome.tbl, but also includes the column N_clusters, the number of fragment clusters in each genome, and N_unique, how many of the clusters are unique to each genome. Lastly, the Cpn.mat is the copy number matrix containing one row for each fragment cluster and one column for each genome. The numbers in the matrix represent how many copies of a fragment cluster can be found in a specific genome. This number is often 0, and therefore the matrix is stored as a sparse matrix.

Making an RMS object for the millions of RMS fragments in HumGut would take a lot of computer resources in memory space and time. For every unique RMS fragment added, their distance to other fragments is calculated, leading to an exponential time algorithm, $O(n) = n^2$. Therefore, the

running time of *RMSobject()*-function on all HumGut genomes and their RMS fragments was estimated using polynomial regression, shown in formula 2, on previously found run times.

$$(2)\ f(x) = c_0 + c_1 x + c_2 x^2$$

Where,
f(x) = Expected running time of *RMSobject(),*
c = a set of coefficients,
x = number of RMS fragments

The resulting regression equation is fitted to the run times of *RMSobject()* on 10, 50, 100, 500, 1000, 5000 random genomes in HumGut. This equation is then used to extrapolate the running time of making an object for the over 30 600 genomes in HumGut, however, it should be noted that this is just a prediction and there are risks by using extrapolation in regression. Consequently, alternative approaches in making an RMS object of a data set as big as HumGut had to be investigated, in case the prediction was far off.

2.4 Assigning NCBI Species and Genus to the Genomes

In further analysis, there was interest in looking into if dividing HumGut into subgroups based on taxonomic rank could be an adequate approach to save computational resources. Therefore, the genomes' taxonomic ranks species and genus were assigned. Genus was assigned instead of family, as it led to fewer and larger subgroups making further analyzations computationally easier. They were assigned by using the genome's NCBI tax ID and the two dmp-files *names* and *nodes* from the NCBI taxonomy database (Sayers *et al.*, 2019; Schoch *et al.*, 2020). The two dmp-files describe a taxonomic tree, and they are regularly updated as taxonomic names and classifications are added, removed, or changed in the database. The names dmp-file contains the names of each NCBI tax ID. The nodes file has several columns of information, where each row represents a node in NCBI's taxonomic database. One of the columns contains the parent node of the node in question in each row, and by using this information taxonomic branches can be found – from leaf to root. Four new columns were added to the HumGut data frame: the genome's species, the species' NCBI tax ID, the genome's genus, and the genus' NCBI tax ID. These columns were then

filled by using the names and nodes files and the functions *branch_retrieve()* and *branch_taxid2name()* from the *microclass* package (Snipen, 2020).

The taxdump.tar.gz archive file, which is publicly available on NCBI, was downloaded (Schoch, 2020; Sayers, 2019). It contains several taxonomy files, but only the *names* and *nodes* dmp-files were used in this thesis. The two selected files were read into R as tibbles using *read_names_dmp()* and *read_nodes_dmp()* from the *microclass* package (Snipen, 2020). The genomes in HumGut were then iterated through, one row at a time, to fill the four new columns. Each iteration started by finding the genome's taxonomic branch using the *branch_retrieve()* function from the *microclass* package. The function takes a genome's NCBI tax ID and the nodes tibble as input and returns a list containing the genome's branch from leaf, input tax ID, to root. The elements under the species and genus rank in the list were selected, and their tax IDs were added to their respective columns. Furthermore, the names of each genome's species and genus rank were collected using the *branch_taxid2names()* function and added to the respective name columns. The function takes the *names* dmp and a vector of tax IDs as input and returns a vector containing the tax ID names.

Several tax IDs in HumGut were outdated and were replaced with updated IDs using NCBI's taxonomy database (Schoch, 2020; Sayers, 2019). Furthermore, several genomes had either no species or no genus rank. These genomes were excluded since further analysis involved dividing HumGut into smaller subsets based on genus. In total, 7410 genomes were excluded. At this point HumGut has 23204 genomes and 16 genome features in further analysis.

2.4.1 Making RMS Objects Based on NCBI Assigned Genera

Due to potential bottlenecks in computer resources when making an RMS object of all RMS fragments in HumGut, an alternative approach of clustering the fragments in subgroups was investigated. HumGut was divided into groups based on genome's genus. Smaller genera were filtered out to avoid groups that were too small for genome clustering. Directories for each of the remaining genera were made, and their genomes' RMS fragment fasta files were copied into them.

The RMS objects were made by iterating through a list of the genera and making an object per genus the same way and using the same values as in section 2.2.1. This resulted in an RMS object which was saved as a single R object (.rds) at the end of each iteration, one per genus.

2.4.2 Performing Genome Clustering on Genera

In RMS, along with all other sequencing methods, it is difficult to separate between genomes that are too similar. Similarity between genomes, in RMS, is the similarity between the columns in the cpn matrix in their RMS object. If the columns are similar, the genomes contain similar RMS fragments. When sequencing, it can be hard to distinguish to which of the genomes the RMS fragments belong. The genome collection is therefore reduced by clustering together genomes that are too similar based on their RMS fragment content. This was performed on the RMS object for all fragments in HumGut, as well as for the RMS object for each genus.

genomeClustering() from the microrms package, (Snipen, 2021), computes the correlation distance between genomes' copy number matrix and clusters together the genomes that are deemed too similar. The function has the RMS object and a max condition value as input. First, the distance between all genomes in the RMS object is calculated by subtracting the correlation between each genome in the copy number matrix from 1. If genome A has an identical column to genome B in the matrix, their correlation is 1 and the distance is 0. On the other hand, if none of genome A's fragments are found in genome B, the correlation is -1 and the distance is 2. After the distances are calculated, a hierarchical clustering with complete linkage is performed. The resulting dendrogram tree can be cut using a height threshold, producing a unique clustering of the genomes. The largest dendrogram height that results in a copy number matrix with a condition value below the user-specified tolerance is chosen. Note that the condition value is calculated first after the hierarchical clustering is performed. The closer the genomes in the copy number matrix are, the higher the condition value is. The unique clustering of genomes is returned as an updated copy number matrix in the RMS object.

The *genomeClustering()* function returns the RMS object with the updated copy number matrix, with a column per cluster, along with an updated cluster and genome table. The genome table now also contains the column "members_genome_id". The column indicates which of the original genomes have been grouped into each cluster, with one of the original genomes representing the cluster as the cluster centroid. How many genomes are clustered together depends on the maximum tolerated condition value.

Although the genomes are clustered together based on their correlation distance, the final clustering must have a condition value under the user-specified threshold. The lowest theoretically possible condition value is 1, but this is not achievable in practice. The lower the condition value, the harder the clustering is which results in fewer genome clusters, and vice versa with higher condition values. The value is important when using the *rmscols*() from the *microrms* package to estimate the abundance of each cluster in samples (Snipen, 2021). The lower the condition value, the more differentiable the genome-clusters are which results in more stable estimates when estimating the abundance of each cluster. On a more technical level, the *rmscols*() inverts the covariance matrix of the input RMS object's copy number matrix in order to estimate the abundances. If two genomes are too similar, their copy number matrix columns are also very similar. This can lead to the covariance matrix not being able to be inverted, or if it can be inverted the similar columns can lead to very unstable results leading to poor estimates. The effect of this is measured by the condition value (Snipen, 2021). The reason the condition value is not directly used to cluster the genomes is it would mean computing the condition values of all linear combinations of clusters. This is quite computationally heavy and would drain a lot of resources, time being one of them. However, it should be noted that using correlation distance does not necessarily guarantee the problem of too similar genomes is resolved. Two uncorrelated genomes might combine into something very correlated to a third genome in the *rmscols*() function, however, in reality this rarely happens (Snipen, 2021).

2.5 Analyzing an External Data Set for *Bifidobacterium* Genomes

In further investigations, there was an interest to see how RMS with HumGut performed on real-world samples from the human gut. The samples were processed and mapped to the genome-clustered RMS object for *Bifidobacterium*, and the abundances were estimated. These estimated abundances were then used to see whether a correlation of the genus content within mother-child pairs could be found.

A data set from PreventADALL of human gut samples from 16 mother-child pairs, where all children were born vaginally, was provided by Nielsen (2022) (Carlsen *et al.*, 2018). Four samples were taken from each mother-child pair, resulting in a total of 64 samples in this data set. The samples consisted of a stool sample from the mother, a swab of the child's skin straight after birth, a sample from the child's meconium (the child's first stool sample), and a stool sample from the child at 3 months. These samples were sequenced, with a high sequencing depth, into fastq-files by using RMS combined with Illumina paired-end sequencing. The resulting de-multiplexed fastq-files were provided along with a metadata table containing the sample information. Important sample information included a unique sample ID (sample_id), a unique mother-child ID (nnid), and the sample "Age" - meaning what substance the sample contains.

2.5.1 Pre-processing Reads from the External Data Set

The data consisted of paired-end Illumina reads in fastq file format that needed to be down-sampled and pre-processed to form a sequence in a fasta file format. VSEARCH was utilized to pre-process the reads in each sample's fastq files, R1 and R2, by quality filtering, merging, trimming primers, and finally adding the processed reads to one fasta file per sample. This would take a lot of time to do for each sample with their high sequencing depth, so the samples were first down-sampled to 1 000 000 reads per sample. Due to the Illumina R1 and R2 fastq file format, meaning the reads are in no particular order and each read takes up 4 lines in the file, the first 4 000 000 lines in the sample fastq file were selected as down-sampled data.

After the down-sampling, a pre-processing shellscript was run using 64 array-jobs, meaning the script was run 64 times – once for each sample. The pre-processing shell script followed the example script found in the tutorial, under "Processing reads", in the *microrms*-package Readme.md (Snipen, 2021). First, reads that were under 30 bp and exceeded the 0.02 error rate were filtered away. The reads were then merged, however, both merged and unmerged reads were kept. Lastly, primers were trimmed from the ends of the reads, before the reads were all added to a fasta-file named after the sample ID. The script resulted in one gz-compressed fasta file per sample.

2.5.2 Mapping Reads to *Bifidobacterium*

The r*eadMapper()*-function from the *microrms* package, (Snipen, 2021), was used to map each sample's pre-processed reads to the *Bifidobacterium* genome-clustered RMS object. The input arguments of the function are the RMS object you want to map the reads to, the sample's fasta file, the VSEARCH executing command, and the identity threshold for mapping the reads. The function returns the RMS object with an added readcount matrix, and the total number of reads (read_total) and number of reads mapped to the object (read_mapped) added as columns in the sample table (sample.tbl).

All sample fasta files were mapped to the *Bifidobacterium* genome RMS object with an identity threshold of 0.99. VSEARCH was run as a singularity container through Orion, using the *u_global* function to map the sample's reads to the fragment cluster centroids in the Cluster.tbl of the genome RMS object. The returned readcount matrix had one column for each sample and one row for each fragment cluster. Further, the readcounts were not normalized for length biases that might have arisen from the RMS amplicons, as there were no prominent biases. The returned RMS object, with the added readcount matrix, was used to estimate *Bifidobacterium* abundances in the samples.

2.5.3 Estimating *Bifidobacterium* Abundances

The *Bifidobacterium* abundances were estimated by using the *rmscols*()-function from the *microrms* package (Snipen, 2021). The function estimates the fraction of each genome in a sample, given the read counts and copy number for each amplicon cluster. This is done by using a Constrained Ordinary Least Square estimation (Snipen *et al*., 2021). In short, the function looks for linear combinations of genome abundances that best explain the observed readcounts in a sample given the cpn matrix.

The *Bifidobacterium* genomes clustered RMS object, with its cluster copy number matrix (cpn) and readcounts matrix (readcount.mat), was used as input. Default settings were used for the features trim, fraction of extreme readcounts to be discarded when fitting a linear model, and reltol, the relative tolerance for the iterative constrained least square search. The default setting for trim is 0, and 10^{-6} for reltol. The function returns an abundance matrix, abd.mat, with one row per genome found in the cpn.mat and one column per sample found in the readcount.mat. In other words, the matrix shows the estimated relative abundance of all genomes in the corresponding sample.

Lastly, there was an interest to see if the samples from a mother correlated to the samples of the corresponding child. This is of interest to see if the RMS method manages to detect vertical transmission of *Bifidobacterium* in the external dataset. The correlation between each child's 3-month sample and all the mother samples, a total of 16x16 correlations, was calculated in order to see if the child's sample shows a stronger correlation to its true mother compared to the other mothers in the data set. A matrix was made with a child's 3-month sample correlation per row, and the mother per column. The correlation was calculated using the *cor*()-function in R and its default method of Pearson correlation (R Core Team, 2021). The Pearson correlation coefficient can be between 1 and –1, where 1 means the samples are positively correlated and –1 means the samples are negatively correlated. A correlation of 0 means there is no correlation between the samples.

3. Results

3.1 The HumGut Data

The HumGut dataset contains 30 614 genomes and 1588 species as of October 2022. After making the RMS fragments for each genome, genome information across HumGut was studied to gain insight into their properties. Some of this information is gathered and displayed in table 2.

Table 2: Genome Information Across HumGut. *It includes the average number of RMS-fragments, expected number of RMS-fragments, GC-content, and genome size per genome. The expected number of RMS-fragments is calculated by using formula 1 in section 2.3, and it depends on the genome's individual GC-content and genome size. The table also includes the sum of all RMS-fragments, expected sum of RMS-fragments, and the total number of species for all 30 614 HumGut genomes. Lastly, it contains the sum of all RMS clusters after fragment clustering of all RMS-fragments in HumGut.*

Genome Information		
	Averages Across HumGut	**Standard Deviations**
RMS-fragments	466	415
Expected RMS-fragments	469	255
GC-content	0.497	0.09
Genome size (bp)	1 988 419	784 494
	Sum Across HumGut	
RMS-fragments	14 279 334	
Expected RMS-fragments	14 386 198	
	Sum in RMS object	
RMS clusters	5 581 004	

Although the average amount of RMS-fragments per genome looks as expected in table 2, with there being 466 fragments per genome when it was expected to be 469, the standard deviation of 415 shows the number of fragments varies a lot between genomes. It varies more than the calculated expected number of fragments had it been random DNA, which displays a standard deviation of 255. Further, the GC-content seems to be about 50% in genomes as the standard deviation is low at 0.09. The genome size varies as well, with the average size being 1 988 400 bp, with a standard deviation of 784 500 bp. Lastly, the table shows there are over 14,28 million RMS fragments in the entire HumGut database which is just about 100 000 less than expected. After clustering the fragments based on 99% similarity, there were 5,58 million RMS fragments - reducing the number of fragments by ~2.5-fold.

The next step of clustering the HumGut genomes, that were too similar based on their RMS fragment content, proved difficult. Clustering involved computing the correlation distance between all genomes' RMS fragment content. Although the fragment-clustering led to a reduction in number of fragments by ~2.5 fold, the clustering was still deemed too computationally heavy. Even with 1,5 terabytes memory reserved on Orion, *genomeClustering*() was not able to cluster all the genomes in the fragment-clustered RMS object in HumGut.

Since the genome-clustering proved to be too computationally demanding, the genomes were ordered into subgroups based on their genus. Only genomes with an assigned genus were kept, resulting in 23 204 genomes in a total of 324 different genera. All the subgroups contained a varied number of genomes, and 84 of them contained only one genome. Since further analysis involved clustering the genomes within each genus, there must be several genomes in the genus in order to have genomes to cluster. Therefore, genera containing 10 or less genomes were filtered out, resulting in a total of 22 595 genomes in HumGut. These genomes were distributed amongst 127 genera used in further analysis.

The fragment distribution within different genera was visualized, where smaller (*Weisella, Eggerthella)* and larger (*Streptococcus, Clostridium, Faecalibacterium*) genome abundant genera were chosen along with genera often associated with being present in the human gut microbiome (*Bacteroides, Bifidobacterium)* (figure 1).

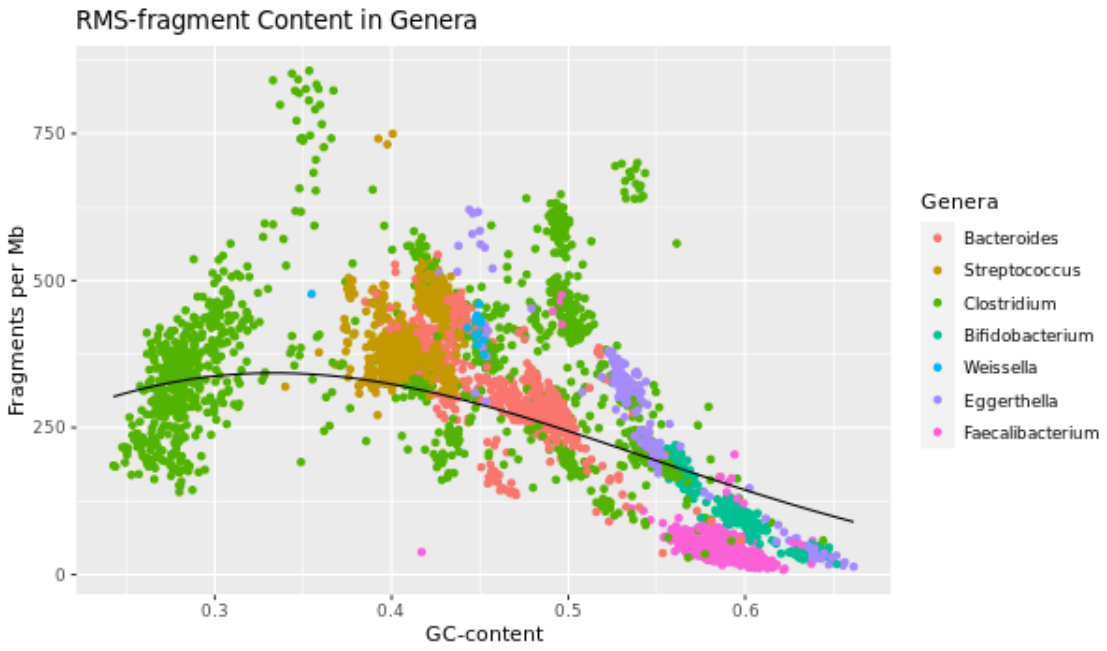


Figure 1: RMS-fragments per Mega-bases in Selected Genera. *The plot shows RMS-fragments per mega bases (Mb) along the y-axis, and the genome GC-content along the x-axis. Each dot represents a genome, and its colour which genus it is from. The black line represents the expected number of fragments per Mb depending on the GC-content*

The number of fragments in each genome can vary a lot between genera, and in a few genera, they also vary within the genus (figure 1). Most genera, like *Faecalibacterium* and *Bifidobacterium* with high GC-content and relatively few RMS fragments, seem to show consistency in both number of fragments and GC-content in their HumGut genomes. By contrast, *Clostridium* stands out the most with a wide range of GC-content in its genomes, and very varied amounts of RMS fragments. The genus also shows it has the most genomes in the plot, showing that the genus has quite a large presence in HumGut compared to the other genera. Further, none of the genera seem to follow the trend of the expected number of fragments. *Faecalibacterium* and *Bifidobacterium* show, mostly, fewer fragments than expected, while the other genera show, mainly, more.

3.2 RMS Objects for HumGut Genomes

3.2.1 All Genomes

Due to there being over 14 million RMS fragments in HumGut, it was expected that measuring the similarity between fragments would take a lot of computer resources like time. Therefore, the run time of *RMSobject*() on all fragments was estimated beforehand. The running time was measured on randomized subgroups of genomes in HumGut, and a 2^{nd} degree polynomial regression was fitted to the data (formula 2). The regression line was then used to extrapolate the runtime of all RMS fragments in HumGut, shown in figure 2.

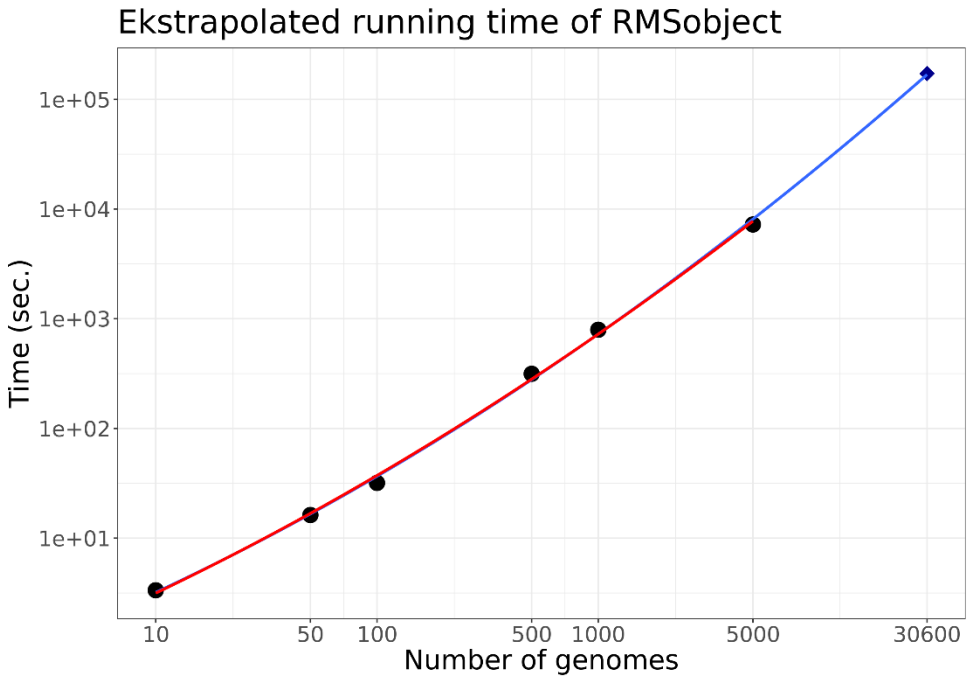


***Figure 2: Extrapolated running time of RMSobject*().** *The actual running time on subgroups of 10, 50, 100, 500, 1000 and 5000 randomized genomes are shown as dots in the plot. A regression line, shown in red, is fitted to them. The regression line from 5000 genomes to 30 600, in blue, shows the extrapolated running times – with the running time of 30 600 genomes marked with a blue diamond point. Note both the x and y axis are log_{10}-transformed.*

Figure 2 shows a regression line that fits well to the running times of the HumGut subgroups. The extrapolated runtime of *RMSobject*() on all RMS-fragments in HumGut is estimated to take around $10^{5.23}$ seconds, which is the equivalent of 2 days. A 95% confidence interval of the predicted running time resulted in the interval: [157500, 18600] in seconds, which equivalates to [1.82, 2.15]

days. Subsequently, the *RMSobject()* was run on all RMS-fragments in HumGut, and the running time and memory used for the job was noted.

- Predicted running time of *RMSobject()* on all RMS-fragments in HumGut: 2 days
- Actual running time: 11,5 days
- Maximum amount of memory used at any time during the job: 29.5 Gb

Since the running time deviated by more than 9 days than predicted, alternative methods to run RMSobject on the HumGut genomes was investigated – resulting in making several RMS objects, one per genus, instead of making one big object for all HumGut fragments.

3.2.2 RMS Object per Genus

The RMS fragments for all genomes within a genus were clustered together the same way as for all the fragments in HumGut. This resulted in 127 fragment-clustered RMS objects, one per genus. This was expected to take a lot less time and be less computationally heavy compared to the clustering of all fragments in one go, as the number of RMS fragments is considerably smaller in the subgroups. Clustering of all genus RMS objects, separately, took a total of 4.07 hours and 2.7 Gb of memory. There was an interest to see if the reduction of RMS fragments after clustering in the genus subgroups resulted in fewer fragments than if HumGut was divided into subgroups at random.

Table 3: Reduction Factors of RMS Fragments after Clustering in Both Randomized Genomes and Genomes within a Genus. *The table includes the reduction factor for subgroups of HumGut with 10, 50, 100 and 500 genomes, while the reduction factor for all fragments across HumGut is noted at the end. The reduction factor was calculated by 1 - (number of RMS fragments after clustering / number of fragments before clustering), so a higher reduction factor corresponds to a stronger reduction in number of RMS fragments. The genomes within a genus were random genomes from the genus Streptococcus, except the 10 and 100 genome group that were from within a species.*

Number of Genomes	Reduction Factor for Randomized Genomes	Reduction Factor for Genomes within a Genus
10	0.01	0.40*
50	0.05	0.20
100	0.05	0.79**
500	0.12	0.52
1000	0.18	0.63
All genomes in HumGut		0.61

* The 10 genomes were from *Streptococcus vestibularis*
** The 100 genomes were from *Sutterella wadsworthensis*

There seems to be a stronger reduction in RMS fragments when clustering within a genus than if the clustering were to happen at random according to table 3. The highest reduction factor for the randomized genomes is 0.18 for a group of 1000 genomes. This is lower than the lowest reduction factor for the genomes within a genus which is 0.20 for 50 genomes. The highest reduction factor is shown for the 100 genomes from the *Sutterella wadsworthensis* species, indicating that the RMS fragment clustering could be higher within species than genera. Lastly, the reduction factor for all RMS fragments in HumGut was 0.61 which is on par with the reduction factors within a genus.

Further, the clustering of similar genomes in each genus was done directly in an R script and took just over an hour. Almost all genera were able to cluster to a condition value of 10, except the genera *Clostridium* and *Eggerthella* which were only able to cluster with a condition value of 100. There is an interest to see whether the genomes clustered together were mainly of the same species, or if the genomes were clustered together across species within the genus.

Figure 3: Number of Species within the Different Clusters in Streptococcus. *Each plot displays the number of species within a cluster, along the x-axis, and the number of clusters containing the different amounts of species along the y-axis. The plots show, respectively, Clostridium, Eubacterium and Streptococcus. Note that the different genera have different amounts of genomes and therefore different amounts of clusters displayed in the plot.*

Mainly 1 or 2 species seem to be represented in most of the genera's genome clusters (figure 3). A cluster contains at most 4 species in *Clostridium*, 3 in *Eubacterium* and 6 in *Streptococcus.* These clusters only make up a small fraction of the total genome clusters. It is not possible to say, based on figure 3, if the composition of species within the genera is what leads to most genome-clusters containing one or two species. If one species makes up the majority of a genus in HumGut, it is expected that there is a bias towards genome-clusters containing only one species.

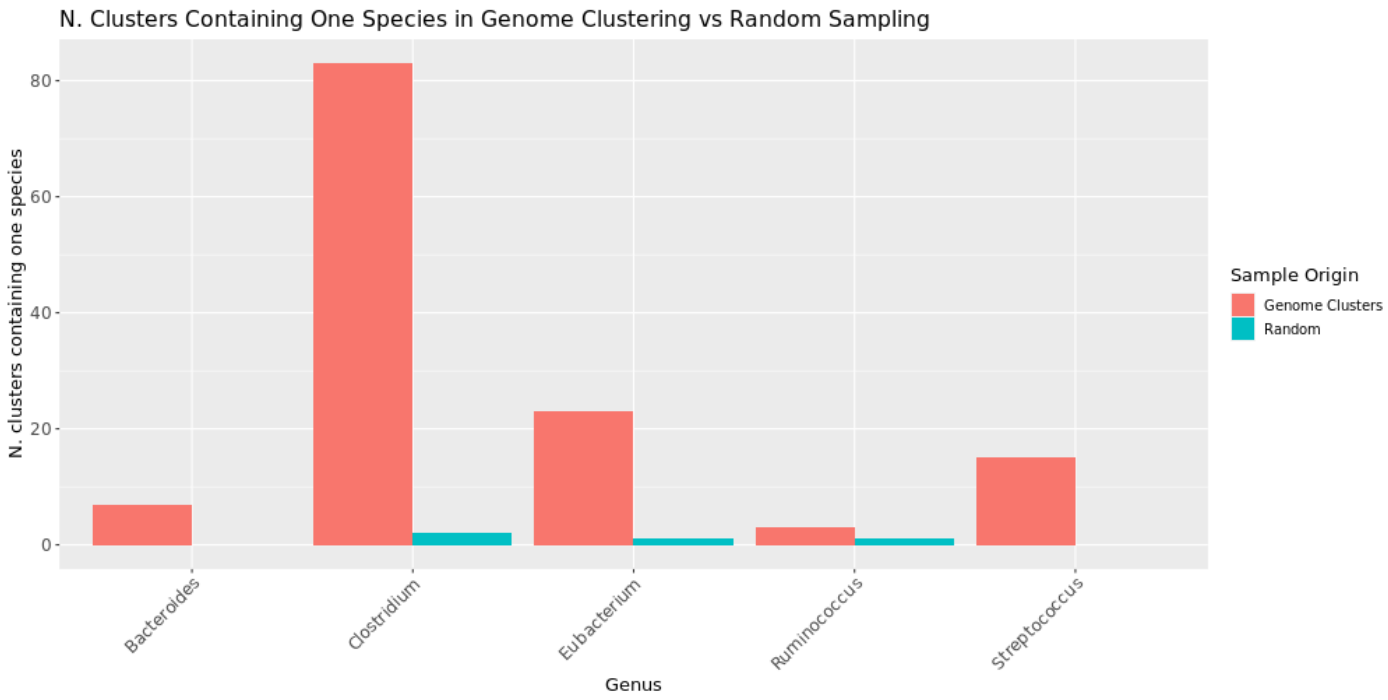


Figure 4: Number of Clusters Containing One Species in Genome Clusters vs. Random Sampling. *Different selected genera are along the x-axis, and the number of genome groupings that contain genomes of only one species are shown along the y-axis. The red bars show the number of genome clusters that contain genomes of only one species in the different genera (based on the genus' genome clustered RMS object). A random sampling was performed, imitating the different genera's number of clusters and their respective genome content. So i.e., if Eubacterium had a genome-cluster containing 3 genomes within the same species, a random sampling of 3 genomes in all Eubacterium genomes in HumGut was performed. If the random sampling returned genomes of the same species, 1 was added to the genus' random count shown in the blue bars. This was done for all genome-clusters containing one species in all the genera displayed in the plot.*

There are more genome clusters than random samples that consist of only one species (figure 4). In *Bacteroides* and *Streptococcus,* none of the random samples showed groupings of the genome containing only one species. *Clostridium* and *Eubacterium* showed very few random samples that

contained the same species compared to the number of genome clusters. *Ruminococcus* displayed the lowest ratio between genome clusters and random samples of the selected genera.

3.2.3 Reduction Factor in RMS Clustering vs. Genome Clustering within Genera

After the division of HumGut into subgroups based on the genomes' genera, there were 22 595 genomes in the dataset. This was reduced to 2270 genome-clusters after genome clustering, making a reduction factor of 0.90 when calculated the same way as in table 3. Further, the reduction factors of both the RMS fragment clustering and genome clustering were calculated. The results for selected genera are presented in figure 5, and the selected genera, along with some of their genome information, is displayed in table 4.

Table 4: Selected Genera and Some of their Genome Information. *The table includes all selected genera along with their number of genomes, total number of RMS fragments, mean GC-content, mean genome size (bp), and why they were selected.*

Genus	*N. genomes*	*N. RMS fragments*	*Mean GC-content*	*Mean genome size (bp)*	*Why they were selected*
Clostridium	1458	913 496	0.418	1 784 474	High amount of genomes & RMS fragments
Streptococcus	1084	770 723	0.408	1 780 686	High amount of genomes & RMS fragments
Faecalibacterium	2065	183 533	0.589	2 005 077	High amount of genomes
Collinsella	2227	321 745	0.601	1 813 042	High amount of genomes
Bfidobacterium	328	55 067	0.595	1 564 584	Often linked to the human gut*
Bacteroides	674	708 946	0.464	3 234 251	Often linked to the human gut*
Eubacterium	593	436 794	0.398	1 755 257	Often linked to the human gut*
Enterobacter	49	51 334	0.555	4 698 587	Small genus with few genomes
Weissella	13	10 875	0.442	1 979 807	Small genus with few genomes
Olsenella	54	1620	0.656	1 644 594	Small genus with few genomes

* (Favier *et al.*, 2002)

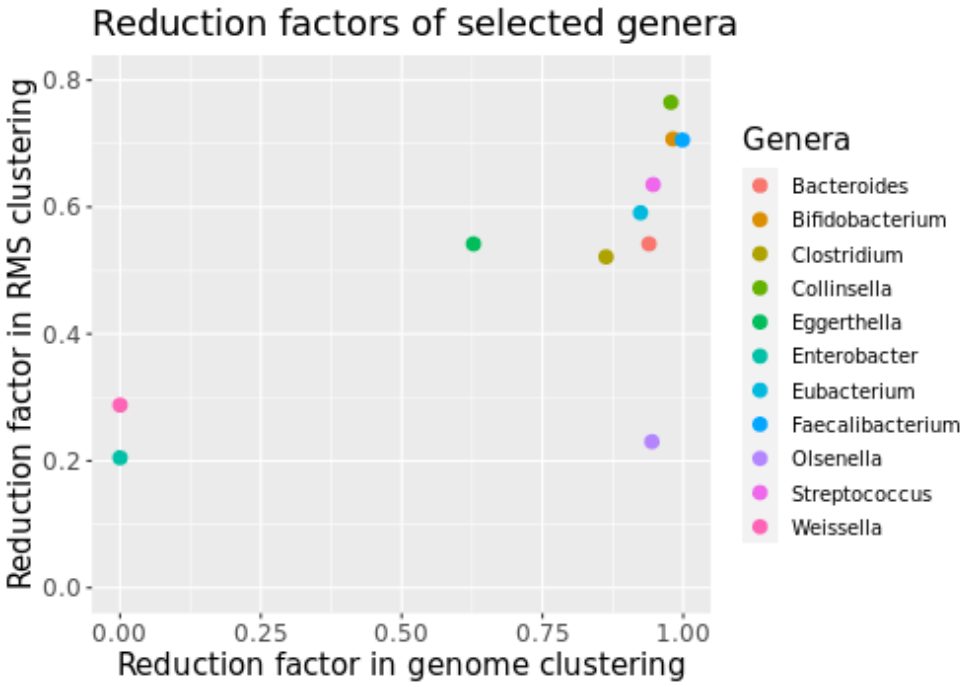


Figure 5: Reduction Factors of both RMS Fragments and Genomes by Genera after Clustering. *The reduction factors of RMS fragments are displayed along the y-axis, and the reduction of genomes within the genera after genome clustering along the x-axis. The reduction factors were calculated by 1 - (number of genomes or fragments after clustering / number of genomes or fragments before clustering).*

Firstly, the least genome abundant genera of *Weissella*, *Olsenella* and *Enterobacter* showed the least reduction in RMS clustering (figure 5). These are also the genera that contain the least amount of RMS fragments to begin with (table 4). *Collinsella*, the most genome abundant genus in HumGut, showed the largest reduction in RMS fragments. The genome abundant *Faecalibacterium* and *Streptococcus* also showed a high reduction of RMS fragments, however, note that *Bifidobacterium* was the genus that showed the next highest reduction after *Collinsella*.

Finally, *Weissella* and *Enterobacter* show no reduction in the number of genomes after genome clustering. They genera contain, respectively, 9 species and 4 strains and 42 species and 7 strains. *Olsenella*, on the other hand, shows a strong reduction of 90%. Both *Eggerthella* and *Clostridium*, the two genera clustered by a higher condition value, show reductions of ~62% and ~86%, respectively. The rest of the genera show a reduction > 90%.

3.3 The *Bifidobacterium* Data

3.3.1 The *Bifidobacterium* Genome Clustered RMS Object

Reads in the samples from the mother-child external data set were mapped to the *Bifidobacterium* genome clustered RMS object. To begin with, 17 different species, 306 strains and 17 subspecies of *Bifidobacterium* are found in the HumGut database. There are 328 *Bifidobacterium* genomes in HumGut containing a sum of 55 067 RMS fragments, which are reduced to 16 124 centroid fragments in the RMS object after fragment clustering. Further, the genomes are reduced 6 genomes after the genome clustering with a condition value of 10, which corresponds to a reduction factor of >98%. The correlation distance between the centroid genomes were calculated and displayed as a dendrogram (figure 6).

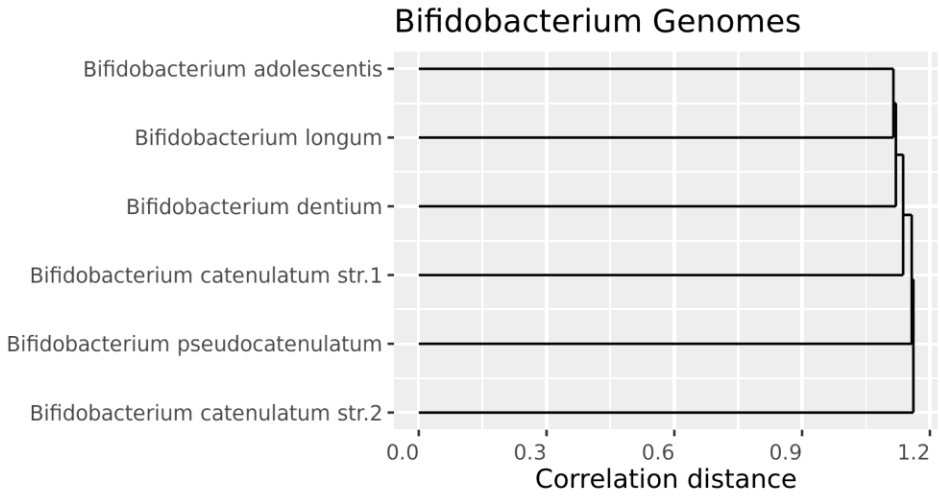


Figure 6: Dendrogram of the Bifidobacterium Centroid Genomes after Genome Clustering. *The correlation distances between the genomes are shown along the horizontal plane. The different centroid genomes, representing their genome cluster, are listed along the vertical plane.*

The correlation distances between the 6 genomes are all above 1 when the genomes were clustered to a maximum condition value of 10 (figure 6). The genomes are from 5 different species, and two of the genomes are even different strains of the *Bifidobacterium catenulatum* species. Further, the condition value of the cpn matrix to the genome clustered RMS object was calculated to 4 using

the function *conditionValue()* from *microrms* package (Snipen, 2021). The reads from the samples of the mother-child data set were mapped against these 6 genomes.

3.3.2 Mapping reads to the *Bifidobacterium* Genus

The down-sampled and pre-processed reads from the 64 mother-child samples were mapped to the *Bifidobacterium* genome clustered RMS object (figure 7).

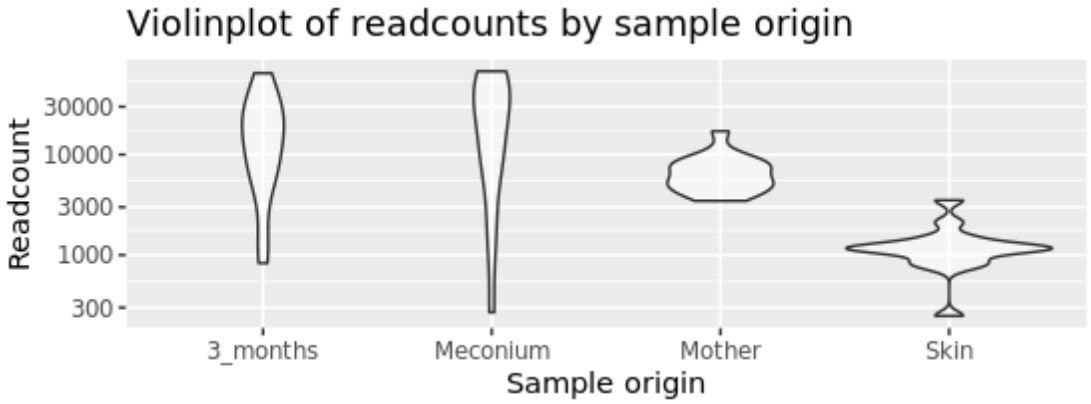


Figure 7: Violinplot of the Average Porportion of Readcounts Mapped to Bifidobacterium. *The plot displays the average relative proportion of reads by sample origin. There are 16 samples per sample origin.*

The average readcounts by sample origin showed the highest relative proportion of *Bifidobacterium* reads are found in the samples taken from the child after 3 months, with around 1-3% of the reads mapping to the genus (figure 7). The meconium samples show the next highest abundance of reads but has the most variable *Bifidobacterium* content in its samples. Relative abundance of *Bifidobacterium* in the mother samples seem more consistent, with most samples getting 0.3%-1% mapped reads. Lastly, are the skin samples where most samples show 0.1% reads mapped to *Bifidobacterium*. These readcounts were further used to see to which *Bifidobacterium* genome they map to.

3.3.3 Estimated Relative Abundances of *Bifidobacterium*

The readcounts and the RMS object's copy number matrix were used to estimate the abundances of *Bifidobacterium* in the mapped reads. The resulting abundance matrix shows which *Bifidobacterium* genomes the reads map to (figure 8).

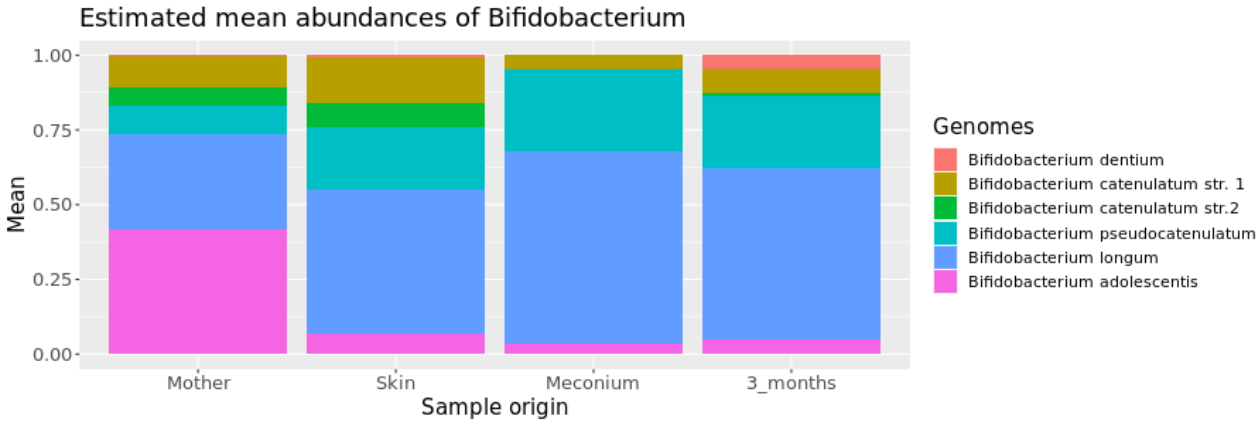


Figure 8: Estimated Relative Mean Abundances of Different Bifidobacterium-genomes. *Of the reads mapped to Bifidobacterium, the figure shows which genome they mapped to, and the relative proportion of reads mapped to each of those genomes. The relative abundances shown are the mean abundances of the samples from each sample origin. The 6 cluster centroid genomes, that the sample fragments are mapped to, are shown on the right. Note two genomes are from B. catenulatum species but are different strains.*

Bifidobacterium adolescentis, *Bifidobacterium longum*, *Bifidobacterium pseudocatenulatum*, and *B. catenulatum str. 1* are found in all mean relative abundances of the sample origins (figure 8). *B. catenulatum str. 2* is found in mother and skin samples, and there is a slight band in the 3-month samples as well. Few reads in the 3-month and the skin samples seem map to *Bifidobacterium dentium*, where the relative proportion of reads mapped to the genome from the skin samples are barely visible. These relative abundances are averages of all mapped reads in all mother-child samples, note that they do not display the potential variation between individual samples.

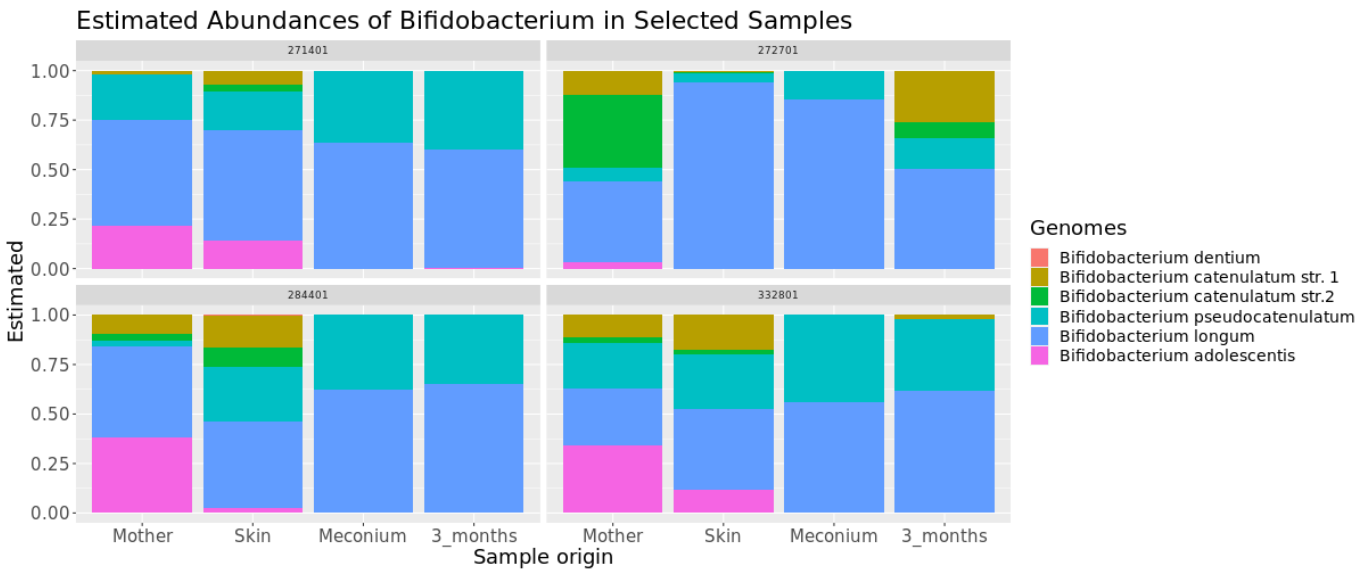


Figure 9: Estimated Relative Abundances of different Bifidobacterium-genomes in Selected Mother-Child Samples*. Of the reads mapped to Bifidobacterium, the figure shows which genome they mapped to, and the relative proportion of reads mapped to each of those genomes. Each plot shows the relative abundance of randomly selected mother-child samples. The 6 cluster centroid genomes, that the sample fragments are mapped to, are shown on the right. Note two genomes are from Bifidobacterium catenulatum species but are different strains.*

The estimated relative abundances of *Bifidobacterium*-genomes in four randomly selected mother-child samples display that there are variations in which genomes the reads map to between samples (figure 9). *B. longum* and *B. pseudocatenulatum*, are consistently found in all mother-child samples. In all meconium samples, and some 3-month samples, these two species are the only ones the sample reads map to. Further, all mother samples and relatively few skin samples contain reads that map to *B. adolescentis*. Lastly, there are variable number of reads that map to the two different strains of *B. catenulatum*, and none of the selected samples that seem to map to *B. dentium*.

There is an interest to see if there are signs of vertical transmission of *Bifidobacterium* genomes in the mapped samples using RMS. The correlations were calculated between a child's 3-month sample and all the mother samples.

Figure 10: Histograms of the correlations between each child's 3-month sample and all mother samples. *Each plot represents the 3-month samples from one child and has the mother-child unique nnid as a title. The Pearson correlation coefficients are shown along the horizontal plane. The frequency of the different correlations, counts, are shown vertically. The black stapled line shows in which bin the correlation to the true mother can be found.*

The correlations, including the correlation between true mother and child, seem to vary a lot (figure 10). There is seemingly no pattern of correlation between a mother and its child's 3-month sample, as both strong positive and strong negative correlations are found. Further, the correlation of a child's 3-month sample and the other unrelated mothers, are spread out showing both positive, negative and no correlation between the samples.

4. Discussion

4.1 The HumGut Data

In this thesis, the use of RMS in combination with the HumGut genome collection was explored to see whether this could be another tool in investigating the microbial composition of the human gut. Previous studies have shown that RMS could obtain a taxonomic resolution to species level, and in some cases even strain level, which is higher than the widely used 16S amplicon sequencing (Snipen *et al.*, 2021). The HumGut collection was chosen as a reference database in this thesis. A potential bottleneck in this method was the size of HumGut with its 30 614 genomes as of October 2022. Consequentially, an alternative method in clustering the entire HumGut genome collection, by dividing the collection into subgroups based on genera, was investigated.

Information gathered about the HumGut genomes' fragment content, in table 2, shows that the number of RMS fragments can vary a lot between the genomes across the data set. Although the averages of expected and actual RMS fragments show a fragment content just beneath 470 fragments per genome, their standard deviations are quite high being, respectively, 255 and 415 fragments per genome. Further, the clustering of all 14.2 million RMS fragments in HumGut by a 0.99 id similarity threshold led to a reduction by ~2.5 fold to 5.58 million fragment-clusters (table 2). This reduction indicates a lot of the RMS fragments are shared between genomes. However, even with the reduction of RMS fragments into fewer fragment clusters, the fragment content was still too high to perform genome clustering of all genomes in HumGut. In the genome clustering function, a distance matrix is made with the distances between all fragments. This is calculated by "unpacking" the sparse copy number matrix from the clustering of the HumGut RMS fragments and calculating the distances, resulting in a distance matrix with the dimensions 5.58 million x 5.58 million. Even 1.5 Tb reserved memory on Orion was not enough to perform the genome clustering, indicating that this is not an efficient enough method on data sets as large as HumGut. This strengthens the idea of dividing large data sets like HumGut into subgroups in order to cluster them.

The fragment composition within the selected genera subgroups of HumGut, showed a mostly homogenous fragment-density (figure 1). The genome abundant genera *Streptococcus* and *Faecalibacterium*, as well as the smaller *Bifidobacterium* and *Weisella*, show a similar GC-content and number of fragments per Mb within the genus. This could be a good sign for the genome quality being sufficient in HumGut considering most of the genomes are MAGs. It is expected that organisms organized within a genus display genetic similarities, (Gill *et al.*, 2005), and similar genomes are expected to produce a similar RMS fragment content (Snipen *et al.*, 2021). Thus, if the fragment content of genomes within a genus showed major differences, it could have indicated a poor genome quality of the HumGut genomes. On the other hand, a genus that does display a wide variety in fragments per Mb for the same GC-content, and vice versa, is *Clostridium.* However, this difference in fragment-distribution is not necessarily due to poor quality in their MAGs. Several papers claim that *Clostridium* contains several genera, as the genomes are quite diverse and not deemed phylogenetically coherent (Cruz-Morales *et al.*, 2019; Stackebrandt *et al*., 1999). This could indicate that the genomes in *Clostridium* are quite different which leads to the genus' varied fragment distribution in figure 1.

Furthermore, most genera seem to display more fragments than the expected number of RMS fragments with regards to the GC-content and genome size (figure 1). This is somewhat surprising as Snipen *et al.* 2021 found most genomes had fewer fragments than expected in random DNA when using RMS. Again, the quality of the MAGs in HumGut could come into question. MAGs are usually of lower quality; it could be a reason for the genomes displaying more fragments. However, if this were to be the case, it is curious that the fragment-density was so heterogeneous within most of the genera. On the other hand, GC-rich genera, like *Bifidobacterium* and *Faecalibacterium*, show fewer fragments per Mb than expected for the respective GC-content and size. This is probably an effect caused by the restriction enzymes, EcoRI and MseI, having AT-rich target sites.

4.2 RMS Objects for HumGut Genomes

The predicted running time of *RMSobject()* on all ~30 600 genomes, predicted at 2 days, deviated a lot from the actual running time of 11.5 days (figure 2). There can be several factors that have played into this deviation, however, the exact reason for the delay is not known. Firstly, the running time was extrapolated using a 2nd degree polynomial regression on smaller subsets. The extrapolated point, in this case the running time of 30 600 genomes, was a lot higher than the largest of the given subsets, the running time of 5000 genomes, used as a basis for the extrapolation. This increases the risk of imprecise measurement and bringing biases into the prediction. Secondly, the function could become quite computationally heavy somewhere between the 5000 and 30 6000 genomes making the runtime a lot longer than expected. An initial suspicion was a lack of memory set for the task. However, the memory used at any point during the running of the clustering shows the function used at most 29.5 Gb when 100 Gb was reserved. At some point between 5000 and 30 600 genomes, the number of fragments seem to become too much for the VSEARCH algorithm to handle in a time-effective way. Lastly, VSEARCH was not run directly in the shell script submitted to Orion. It was run in the Rscript which was called on by the shell script submitted to the HPC. There could be delays in running time when running the singularity container in such a way.

4.2.2 RMS Object per Genus

Dividing HumGut into subgroups based on the genomes' genus resulted in a much faster running time of the RMS fragment-clustering, and in fewer RMS fragment-clusters compared to dividing HumGut into randomized subgroups (table 3). Similar genomes, like genomes within a genus, are expected to share more RMS fragments resulting in a larger reduction in the number of fragments after clustering. Table 3 exhibits higher reduction factors in RMS fragments for genomes within a genus than for subgroups of random genomes. The highest reduction factor of 0.79 is found in the subgroup containing 100 genomes from the species *S. wadsworthensis.* The highest reduction factor within a genus is 0.63 containing 1000 *Streptococcus* genomes. This supports the fact that genomes within a genus are more similar in RMS content, and that genomes within a species

display even more similarities in their RMS content. From these results, dividing HumGut into genera seems more sensible than dividing it at random.

In this thesis all genomes within a genus were attempted clustered with a maximum tolerated condition value of 10. Only two of the genera were not able to produce genome-clusters with a condition value of 10 or under. The lower the condition value, the harder the clustering is which results in fewer genome clusters. In return, when later estimating abundances of the different clusters in a sample, the abundancy estimates are more stable since the clusters are easier to differentiate. However, the more genomes that are clustered together, the lower the taxonomic resolution becomes. Conversely, a higher condition value would return more genome clusters and a higher taxonomic resolution, but they would be more similar and harder to distinguish than the clusters found with a condition value under 10. This creates a trade-off of either a lower tolerated condition value gaining more stable estimates but at the price of a lower taxonomic resolution, and a higher tolerated condition value gaining a higher taxonomic resolution but with more unstable estimates. If a high taxonomic resolution is of importance, this is an advantage of the RMS method; A higher taxonomic resolution *can* be achieved by tolerating a higher condition value, but then the stability of the abundance estimates should be investigated further. This could be done by, e.g., calculating the correlation distances between the genome clusters in order to see how similar they are. The lower the correlation distances, the more similar they are, and the more unstable the abundancy estimates will be.

Clostridium and *Eggerthella* were the two genera that could not produce a combination of clusters with a condition value as low as 10. With *Clostridium* showing such a diverse number of fragments per Mb and GC-content in their genomes (figure 1), it is rather surprising that none of the genome-cluster combinations can make a condition value of 10 or lower. One would expect a higher correlation distance between genomes displaying different RMS fragment contents, which would return more distinct genome-clusters and a lower condition value. However, the fact that the genomes are clustered based on their correlation distance, and not condition value, could explain why *Clostridium* was not able to achieve a lower condition value. When using correlation distance, the distance between pairs of genomes is calculated. However, two uncorrelated genomes might

be combined to become very correlated to a third genome during the *rmscols*() function, leading to a higher condition value. This is a potential problem in general when clustering based on correlation distances, but previous practical experience suggests that this is problem rarely occurs (Snipen, 2021).

Further, there was an interest to see whether the genomes that clustered together were from the same species or from different species within the genera. This is important as the species compositions of the genome-clusters essentially determine whether a taxonomic resolution down to species level can be achieved. If the clusters mainly comprise of genomes from different species, the taxonomic resolution is compromised when samples are mapped against these clusters. Additionally, if this were the case the quality of the genomes in HumGut could be questioned. It is expected that genomes within the same species are clustered together due to genomic similarities resulting in similar RMS fragments. This was somewhat shown in table 3, where genomes within *S. wadsworthensis* resulted in the highest reduction factor of 0.79 in its RMS fragment-clustering. Figure 3 illustrates that most genome-clusters within the selected genera contain 1 or 2 species, indicating genome-clustering follows the species boundaries within genera. There are clusters that show a higher species content, at most 6 species are in a cluster of the genera shown in figure 3, but they only make up a diminutive fraction of the total genome-clusters. This may not always be the case for all other genera as interspecies similarity within genera can vary.

To further investigate whether clusters tend to align with species, the number of clusters containing only one species were counted, as seen in figure 4. This count must be compared to how species distribute in a random clustering, where groups of genomes are formed randomly. If the random count is as high as the genome-cluster count it would indicate a bias in the genus' species content in HumGut, which would weaken the claim that the genomes clustering mainly follow the species boundary. However, this is not the case in figure 4 where the random counts are barely visible. It strengthens the argument that the genomes tend to cluster based on species, as expected. This could further indicate that a taxonomic resolution down to species level can be gained and that the HumGut MAGs do not seem to be of very poor quality.

4.2.3 Reduction Factor in RMS Clustering vs. Genome Clustering within Genera

The reduction factor in RMS fragment clustering within genera shows a higher reduction in the more genome abundant genera compared to the smaller genera as expected (figure 5). The more genomes within the genera, the more possibilities of shared fragments there are which leads to a likely higher reduction factor in RMS content compared to genome-poor genera. *Bifidobacterium*, however, is not one of the most abundant genera in HumGut but still displays one of the largest reductions in RMS fragment content. This could mean the genomes within *Bifidobacterium* in HumGut are quite similar, sharing a lot of the same RMS fragments.

After the clustering of genomes, most genera show a reduction of >80% in number of genomes (figure 5). The harder clustering using a condition value of 10 has led to fewer genome clusters, which can explain the high reduction factors within genera. This will most likely come at a price with regards to the taxonomic resolution when mapping reads to these genome clusters. It is important to highlight that the taxonomic resolution gained with these genome-clusters are the lowest possible taxonomic resolution for the different genera due to the low condition value. An increase in the condition value will likely lead to lower genome reduction values for the genera.

The 16S amplicon sequencing usually gains a taxonomic resolution on genus level, (Snipen *et al.*, 2021), and a genus level taxonomic resolution here would mean a reduction factor within the genera of 1. That would mean all the genomes within the genus would be clustered together with one genome representing them all as the cluster centroid. Some genera come close to 1 here, like *Faecalibacterium* and *Bifidobacterium* which are GC-rich genera with few fragments (table 4 & figure 5). The fewer fragments in the genera could lead to difficulties in differentiating the genomes based on their RMS content, leading to a lower taxonomic resolution. On the other hand, the smaller genera *Weissella* and *Enterobacter* show a reduction factor of 0, meaning none of their genomes were clustered together in the genome-clustering. All their genomes are represented as their own cluster centroid and are therefore easily differentiable using the RMS method in this thesis. Looking at *Enterobacter* and *Weissella*'s genome information in table 4, they have the most

RMS fragments per genome of the selected genera. Several of the other genera show a genome reduction of around 90%, meaning they can gain somewhat deeper taxonomic resolution than genus level. *Olsenella* is one of these, even though it contains few genomes like *Enterobacter* and *Weissella*. Looking at its RMS fragments, it has considerably fewer fragments per genome compared to the other two genera (table 4). Lastly, *Clostridium* and *Eggerthella* show lower reduction factors, which makes sense as they had a higher tolerated condition number leading to more genome clusters. In summary it seems the more RMS fragments per genome, the easier it is to differentiate between them.

The HumGut dataset consisted of 22 595 genomes that were reduced to 2270 genome-clusters after genome clustering of all the genus subgroups, which corresponds to a reduction factor of 0.90. Most of the selected genera displayed a similar reduction factor to this (figure 5). Since genome-clustering seems to follow the species boundaries by mainly clustering genomes from the same species (figure 3 & figure 4), it can be somewhat expected that this trend continues up the taxonomy tree. This is somewhat supported by the high reduction factors within genera, (figure 5), as genomes within genera cluster more than genomes within random subgroups (table 3). By extrapolating this idea, if the clustering of all genomes in HumGut proved possible, its number of genome-clusters may not be far off from the sum of genome-clusters in all the genera. If it followed a similar reduction factor as for genome clustering within genera, it would result in ~3000 genome-clusters. There are 1588 species within HumGut, so if the reduction factor is not far off, and the maximum tolerated condition value for the genome clustering can be set higher, a taxonomic resolution down to species level seems achievable.

4.3 The *Bifidobacterium* Data

A dataset from the PreventADALL study was provided to this thesis in order to map sample reads and estimate abundances for a specific genus (Carlsen *et al*., 2018; Nilsen, 2022). The *Bifidobacterium* genome-clustered RMS object, clustered with a condition value of 10, was used as *Bifidobacterium* is expected to appear in both mother and child gut microbiomes. The

correlation between estimated abundances of a child's 3-month samples and all mothers were calculated to see if there were any signs of transmission of *Bifidobacterium* between a mother and her child.

4.3.1 The *Bifidobacterium* Genome Clustered RMS Object

The >98% reduction in *Bifidobacterium* genomes after genome clustering is not all that unexpected. Previously it was shown that the RMS-fragment content in *Bifidobacterium* genomes is lower than expected and that the genomes are GC-rich (figure 1). Further, the high reduction factor in their RMS fragment clustering could indicate that their RMS fragment content was similar in the genomes (figure 5). A combination of a low RMS-fragment content and their strong fragment clustering could explain the clustering of genomes from 328 genomes to 6 genome clusters. The resulting 6 genomes seem to be very differentiable with a condition value of 4 and large correlation distances (figure 6). 5 of the 17 species were represented as genome cluster centroids, and two of the clusters even contained two different strains of *B. catenulatum* as genome cluster centroids. In other words, the two strains are different enough in RMS content that they are differentiable by this method. However, as 11 of the species are not represented, it is not expected that the genome-clusters only contain one species. Most likely several species have clustered together in the genome-clusters, and as a result the RMS method cannot differentiate between which of the genomes within the cluster a read maps to. This indicates a low taxonomic resolution for *Bifidobacterium* using the RMS method.

4.3.2 Mapping reads to the *Bifidobacterium* Genus

As the reads are only mapped to the *Bifidobacterium* genome clustered RMS object, the mapped readcount will not say anything about what proportion of the entire sample *Bifidobacterium* genomes make (figure 7). So even though the proportion of reads that map to *Bifidobacterium* are low (0.3-4%) in the different samples, it does not mean there are few *Bifidobacterium* genomes in the samples. Since there are on average quite a lot fewer RMS fragments in the genera (figure 1), it will take fewer reads to account for an entire *Bifidobacterium* genome compared to genomes in other genera. If one wants to say something about the proportion of *Bifidobacterium* genomes in

the sample, this fragment-bias must be adjusted for to get a more realistic account. Ideally, the reads should be mapped to the genome clustering object for the entire HumGut to get the relative abundancy of *Bifidobacterium*. Currently we do not have such an object since the genome-clustering of all HumGut genomes required too many computing resources.

Further, the readcount bias for *Bifidobacterium* genomes is the same for all samples. This means comparing readcounts between samples from the different sample origins is possible. The readcounts show that the 3-month samples display the largest proportion of *Bifidobacterium* genomes, followed by meconium, mother and lastly skin samples (figure 7). This was also found by Killingstad 2021, where amplicon sequencing of the *ClpC* marker gene for *Bifidobacterium* was performed on the same samples. This is another indication that the HumGut *Bifidobacterium* genomes are of good quality, although being MAGs. Lastly, the amount of *Bifidobacterium* in the human gut is expected to lessen with age, (Odamaki et al., 2016), which seems to coincide with the readcounts in the mother samples being lower than for the 3-month and meconium samples.

4.3.3 Estimated Relative Abundances of *Bifidobacterium*

The estimated relative abundances of the 6 genome clusters in the mapped reads show that *B. longum* is the species most of the sample reads mapped to (figure 8 & 9), especially in meconium and 3-month samples. This coincides with literature, claiming *B. longum* are one of the first microbes that colonize the human gut (Diaz *et al.*, 2021; Yao *et al.*, 2021). The mother samples also show relatively many reads mapped to *B. adolescentis*. The species is known to have strains that specifically colonize the gut of adult individuals, which would explain the lower abundances of the genome-cluster in the other samples (Duranti *et al*., 2016). Although the reads mapped to these genome-clusters, several of the reads will most likely be of other *Bifidobacterium* species and strains that map to other genomes in the genome-clusters than these cluster representatives. However, when further investigating the correlation between mother and child samples, the estimated abundances of the genome-clusters are more stable.

The correlation data shows no trends of vertical transmission of the *Bifidobacterium* genome-cluster content from true mother to the child's 3-month sample (figure 10). The correlation of abundances between a child and its true mother almost seems random, with the lowest correlation being below –0.6 and the highest above 0.7. Additionally, the correlation of a child's sample to the other mothers seem to vary just as much. If the samples were mapped against *Bifidobacterium* clustered at a higher condition value with more genome-clusters, maybe a trend in correlations could be noticed. It should also be noted that the mother samples were taken during their 18th week of pregnancy, and the mother's gut microbiome could change drastically in between that time and when the child is born. The Prevent-ADALL study was not initially intended for studies of vertical transmissions between mother and child (Carlsen *et al*., 2018). Lastly, most of the *Bifidobacterium* could also simply not be transmitted vertically. Killingstad 2021, also found no association between mother and 3-month samples for several *Bifidobacterium* species. Only *B. longum* showed an association between the mother and the 3-month child samples.

4.4 Conclusion and Future Work

In this thesis the main aim was to investigate the taxonomic resolution RMS can potentially gain using HumGut as a reference database. Unfortunately, it was not possible to obtain a genome-clustered RMS object for all genomes in HumGut as it proved to take too many computing resources than available. Therefore, HumGut was divided into subgroups based on genomes' genus.

When clustering based on genus, instead of for all genomes in HumGut, the taxonomic resolution gained within the different genera differs. The genera that contain more RMS fragments per genome seemed to gain a higher resolution, with some genera like *Weissella* even gaining a taxonomic resolution down to strain level. Conversely, the genomes that contained few RMS fragments per genome, typically GC-rich genomes like *Bifidobacterium*, tended to gain lower taxonomic resolutions not that much lower than genus level. However, these are the results of genome clustering with a condition value of 10 or below and therefore the lowest taxonomic resolution gained, forming a kind of taxonomic resolution baseline for the different genera using

this method. In further studies, higher tolerated condition values should be tested. A higher tolerated condition value can lead to more genome clusters and higher taxonomic resolutions. Additionally, correlation distances should be investigated to examine the similarities between the resulting genome clusters, which could lead to more unreliable abundance estimates.

The clustering of genomes within genera tends to align with species and this trend could follow up the taxonomy tree. The clustering of all genomes within HumGut may not be far off from the sum of genome clusters in all genera, which would result in ~3060 HumGut genome-clusters. If this were to be the case, and since there are under 1600 species within HumGut as well as the possibility for adjusting for a higher condition value during genome clustering, a taxonomic resolution of HumGut down to species level seems achievable. However, the genome clustering of HumGut must be achieved in order to study this further. Unless the RMS fragment clustering and genome clustering improves in efficiency to be able to handle datasets as big as HumGut, alternative strategies in dividing HumGut into smaller subgroups should be investigated. E.g., as mentioned by Lars Snipen (personal communication, December 12, 2022), genome groups that share no RMS fragments could be analyzed separately. Since they do not contain any of the same RMS fragments, this should not impact the taxonomic resolution.

The PreventADALL mother-child samples showed no signs of vertical transmission when mapped to the *Bifidobacterium* genome clusters. The correlations of *Bifidobacterium* abundancies between the children's 3-month sample and their corresponding mother’s sample differed a lot, and no correlation trends were observed. This could be due to *Bifidobacterium* having a low taxonomic resolution with the strict condition value of 10. In further work, if a higher condition value is tolerated leading to a higher taxonomic resolution, a trend might be observed. However, it comes at a cost of more unstable abundancy estimates which are very important when looking into possible vertical transmissions of the genus. Furthermore, the use of different restriction enzymes that are less AT-rich could be explored to see if it gains higher taxonomic resolution in GC-poor genera like *Bifidobacterium*.

References

American Society for Microbiology (2003). Microbial Communities: Advantages of Multicellular Cooperation. Microbial Communities: From Life Apart to Life Together. Washington (DC). doi:10.1128/AAMCol.3May.2002

Baker, G. C., et al. (2003). "Review and re-analysis of domain-specific 16S primers." J Microbiol Methods 55(3): 541-555. doi:10.1016/j.mimet.2003.08.009

Breitwieser, F. P., et al. (2019). "A review of methods and databases for metagenomic classification and assembly." Brief Bioinform 20(4): 1125-1136. doi:10.1093/bib/bbx120

Cermak, N., et al. (2020). "Rapid, Inexpensive Measurement of Synthetic Bacterial Community Composition by Sanger Sequencing of Amplicon Mixtures." iScience 23(3): 100915. doi: https://doi.org/10.1016/j.isci.2020.100915

Cruz-Morales, P., et al. (2019). "Revisiting the Evolution and Taxonomy of Clostridia, a Phylogenomic Update." Genome Biology and Evolution 11(7): 2035-2044. doi:10.1093/gbe/evz096

Díaz, R., et al. (2021). "Comparative Genomic Analysis of Novel Bifidobacterium longum subsp. longum Strains Reveals Functional Divergence in the Human Gut Microbiota." Microorganisms 9(9): 1906. doi:10.3390/microorganisms9091906

Duranti, S., et al. (2016). "Evaluation of genetic diversity among strains of the human gut commensal Bifidobacterium adolescentis." Scientific Reports 6(1): 23971. doi:10.1038/srep23971

Favier, C. F., et al. (2002). "Molecular monitoring of succession of bacterial communities in human neonates." Appl Environ Microbiol 68(1): 219-226. doi:10.1128/aem.68.1.219-226.2002

Federhen, S. (2011). "The NCBI Taxonomy database." Nucleic Acids Research 40(D1): D136-D143. doi:10.1093/nar/gkr1178

Gill, S. R., et al. (2006). "Metagenomic Analysis of the Human Distal Gut Microbiome." Science 312(5778): 1355-1359. doi:10.1126/science.1124234

Goodwin, S., et al. (2016). "Coming of age: ten years of next-generation sequencing technologies." Nature Reviews Genetics 17(6): 333-351. doi:10.1038/nrg.2016.49

Handelsman, J., et al. (1998). "Molecular biological access to the chemistry of unknown soil microbes: a new frontier for natural products." Chemistry & Biology 5(10): R245-R249. doi: https://doi.org/10.1016/S1074-5521(98)90108-9

Hess, M. K., et al. (2020). "A restriction enzyme reduced representation sequencing approach for low-cost, high-throughput metagenome profiling." PLOS ONE 15(4): e0219882. doi:10.1371/journal.pone.0219882

Hiseni, P., et al. (2021). "HumGut: a comprehensive human gut prokaryotic genomes collection filtered by metagenome data." Microbiome 9(1): 165. doi:10.1186/s40168-021-01114-w

Hugenholtz, P., et al. (2016). "Genome-Based Microbial Taxonomy Coming of Age." Cold Spring Harb Perspect Biol 8(6). doi:10.1101/cshperspect.a018085

Janda, J. M. and S. L. Abbott (2007). "16S rRNA gene sequencing for bacterial identification in the diagnostic laboratory: pluses, perils, and pitfalls." J Clin Microbiol 45(9): 2761-2764. doi:10.1128/JCM.01228-07

Killingstad, M.-E. (2021). “Colonization of Bifidobacterium in the Human Infant Gut”. Faculty of Chemistry, Biotechnology and Food Science. Ås, Norway, Norwegian University of Lice Sciences. MSc Biotechnology.

Laursen, M. F., et al. (2021). "Bifidobacterium species associated with breastfeeding produce aromatic lactic acids in the infant gut." Nature Microbiology 6(11): 1367-1382. doi:10.1038/s41564-021-00970-4

Liland, K. H., Vinje, Hilde. & Snipen, L. (2021). “microclass: Tools for taxonomic classification of prokaryotes”. *BMC bioinformatics*, *18*(1), 172. https://doi.org/10.1186/s12859-017-1583-2

Liu, M. Y., et al. (2017). "Evaluation of ddRADseq for reduced representation metagenome sequencing." PeerJ 5: e3837. doi:10.7717/peerj.3837

Lødrup Carlsen, K. C., et al. (2018). "Preventing Atopic Dermatitis and ALLergies in Children-the PreventADALL study." Allergy 73(10): 2063-2070. doi:10.1111/all.13468

Makino, H., et al. (2013). "Mother-to-infant transmission of intestinal bifidobacterial strains has an impact on the early development of vaginally delivered infant's microbiota." PLOS ONE 8(11): e78331. doi:10.1371/journal.pone.0078331

Massello, F. L., et al. (2020). "Meta-Analysis of Microbial Communities in Hot Springs: Recurrent Taxa and Complex Shaping Factors beyond pH and Temperature." Microorganisms 8(6): 906. ISSN:2076-2607

Naghoni, A., et al. (2017). "Microbial diversity in the hypersaline Lake Meyghan, Iran." Scientific Reports 7(1): 11522. doi:10.1038/s41598-017-11585-3

National Human Genome Research Institute (2021) The Cost of Sequencing a Human Genome. Available from: https://www.genome.gov/about-genomics/fact-sheets/Sequencing-Human-Genome-cost

Nilsen, M. (2022). The gut microbiota from the general population during the first year of life. Faculty of Chemistry, Biotechnology and Food Science (KBM). Ås, Norway, Norwegian University of Life Sciences. PhD. ISBN/ISSN: 978-82-575-2019-9/ 1894-6402

O'Leary, N. A., et al. (2016). "Reference sequence (RefSeq) database at NCBI: current status, taxonomic expansion, and functional annotation." Nucleic Acids Res 44(D1): D733-745. doi:10.1093/nar/gkv1189

R Development Core Team (2021). R: A Language and Environment for Statistical Computing. Vienna, Austria, R Foundation for Statistical Computing. Available from: https://www.R-project.org

Ravi, A., et al. (2018). "Comparison of reduced metagenome and 16S rRNA gene sequencing for determination of genetic diversity and mother-child overlap of the gut associated microbiota." J Microbiol Methods 149: 44-52. doi:10.1016/j.mimet.2018.02.016

Reinartz, J., et al. (2002). "Massively parallel signature sequencing (MPSS) as a tool for in-depth quantitative gene expression profiling in all organisms." Brief Funct Genomic Proteomic 1(1): 95-104. doi:10.1093/bfgp/1.1.95

Rognes, T., et al. (2016). "VSEARCH: a versatile open source tool for metagenomics." PeerJ 4: e2584. doi:10.7717/peerj.2584

Schmalenberger, A., et al. (2001). "Effect of primers hybridizing to different evolutionarily conserved regions of the small-subunit rRNA gene in PCR-based microbial community analyses and genetic profiling." Appl Environ Microbiol 67(8): 3557-3563. doi:10.1128/aem.67.8.3557-3563.2001

Schoch, C. L., et al. (2020). "NCBI Taxonomy: a comprehensive update on curation, resources and tools." Database 2020. doi:10.1093/database/baaa062

Sims, D., et al. (2014). "Sequencing depth and coverage: key considerations in genomic analyses." Nature Reviews Genetics 15(2): 121-132. doi:10.1038/nrg3642

Snipen, L. (2022) microrms: Processing and analysis of RMS data. Available from: https://github.com/larssnip/microRMS.

Snipen, L., et al. (2021). "Reduced metagenome sequencing for strain-resolution taxonomic profiles." Microbiome 9(1): 79. doi:10.1186/s40168-021-01019-8

Snipen, L. and Liland, K. H. (2021). " microseq: Basic Biological Sequence Handling." Available from: h ttps://CRAN.R-project.org/package=microseq

Stewart, E. J. (2012). "Growing Unculturable Bacteria." Journal of Bacteriology 194(16): 4151-4160. doi:10.1128/JB.00345-12

Wickham, H. (2016). ggplot2: Elegant Graphics for Data Analysis, Springer-Verlag New York. ISBN:978-3-319-24277-4

Wong, C. B., et al. (2020). "Insights into the reason of Human-Residential Bifidobacteria (HRB) being the natural inhabitants of the human gut and their potential health-promoting benefits." FEMS Microbiology Reviews 44(3): 369-385. doi:10.1093/femsre/fuaa010

Yao, S., et al. (2021). "Bifidobacterium Longum: Protection against Inflammatory Bowel Disease." J Immunol Res 2021: 8030297. doi:10.1155/2021/8030297

Appendix

Appendix A: Code for Making the RMS Fragments

A.1: The Shell Script

```
#!/bin/bash

#SBATCH --array=1-307%15                # jobs 1 to ..., maximum 15 at the time
#SBATCH --nodes=1
#SBATCH --ntasks=1                      # The number of threads reserved
#SBATCH --mem=10G                       # Memory reserved
#SBATCH --partition=smallmem,hugemem
#SBATCH --time=7:00:00                  # Runs for maximum this time
#SBATCH --job-name=RMSfrg               # Sensible name for the job
#SBATCH --output=RMSfrg%j_%a.log        # Logfile output here


####################
### Loading modules
###
module load R/4.0.4


##############
### Settings
###
threads=1

#################################################################
### Reading humgut.genome.tbl in blocks of 100.
### bolk_idx is the index of the first genome in the block.

bolk_idx=$((($SLURM_ARRAY_TASK_ID-1)*100)+1))

######################
### Running R script
###
Rscript r_getfrag.R $bolk_idx
```

A.2: The R Script

```
# Importing libraries
library(tidyverse)
library(microseq)
library(microrms)

# Loading the humgut.genomes.tbl
load(<path to humgutproject file>)

# Shellscript-arguments:
input.arg <- commandArgs(trailingOnly = T)
in.arg <- as.numeric(input.arg[1]) # Shell-script argument

# Finding the RMS-fragments and writing to file:
frg <- "fragRMS"
til <- ifelse(in.arg == 30601, 30614, in.arg+99)

for(i in in.arg:til){
  readFasta(file.path(humgut.genomes.tbl$path[i], humgut.genomes.tbl$genome_file[i])) %>%
    getRMSfragments(genome.id = humgut.genomes.tbl$genome_id[i]) %>%
    writeFasta(out.file = file.path(<destination folder>, frg,
                                    humgut.genomes.tbl$genome_file[i]))
}
```

Appendix B: General Code for Making the RMS Objects

B.1: The Shell Script

```
#!/bin/bash

#SBATCH --nodes=1
#SBATCH --ntasks=10                      # The number of threads reserved
#SBATCH --mem=80G                        # Memory reserved
#SBATCH --partition=hugemem
#SBATCH --job-name=frgClustRMS        # Name for the job
#SBATCH --output=frgClustRMS_%j.log   # Logfile output here


####################
### Loading modules
###
module load R/4.0.4

threads = 10

######################
### Running R script
###
Rscript RMSfrg_all.R
```

B.2: The R Script

```
# Libraries:
library(tidyverse)
library(microrms)

# Reading in table:
humgut.tbl <- read.csv(<path to HumGut dataframe>)
# Defining vsearch run-line:
vsearch_exe <- "srun singularity exec /mnt/users/heolsen/Master/Part2/vsearch:2.21.1--hf1761c0_1.sif vsearch "
# Directory of the fragment fasta files:
frg.dir <- <Path to fragment fasta file directory>

RMSobject(humgut.tbl, frg.dir, vsearch.exe = vsearch_exe, threads = 10) -> obj_all

saveRDS(obj_all, file = "RMSobj_all.rds")
```

Appendix C: Assigning NCBI Rank

This script can also be used to assign genus, by replacing “species” by “genus”

```
library(plyr)
library(microclass)

# Loading the names and nodes dmp files from NCBI
read_names_dmp("names.dmp") -> ncbi_names
read_nodes_dmp("nodes.dmp") -> ncbi_nodes

# Loading the humgut table
raw_hum <- read.csv(<Path to HumGut dataframe>, header=TRUE, stringsAsFactors=FALSE)

# Making columns for the taxonomic names and IDs, ready to be filled in
raw_hum %>%
  mutate(species_name = NA, species_id = NA) -> raw_hum


for(i in 1:nrow(raw_hum)){
  temp_ <- branch_retrieve(raw_hum$species_id[i], ncbi_nodes)
  if (!is.na(temp_[[1]]["species"])){
    raw_hum$species_id[i] <- temp_[[1]]['species'][[1]]
    raw_hum$species_name[i] <- branch_taxid2name(raw_hum$genus_id[i], ncbi_names)
  }

}


raw_hum <- na.omit(raw_hum)

saveRDS(raw_hum, file="humgut_species.Rda")
```

Appendix D: General Code for Making the RMS Objects

This script tries to cluster the genomes in the RMS object for all genomes in HumGut. A modified version of it was used to cluster the genomes within a genus, using these scripts as a basis

D.1: The Shell Script

```
#!/bin/bash

#SBATCH --nodes=1
#SBATCH --ntasks=1                          # The number of threads reserved
#SBATCH --mem=80G                           # Memory reserved
#SBATCH --partition=hugemem
#SBATCH --job-name=genomeClustRMS       # Sensible name for the job
#SBATCH --output=genomeClustRMS_%j.log    # Logfile output here


####################
### Loading modules
###
module load R/4.0.4

######################
### Running R script
###
Rscript genomeClust_all.R
```

D.2: The R Script

```
# Libraries:
library(tidyverse)
library(microrms)

# Reading the main RMS object
RMS.obj <- readRDS("RMSobj_all.rds")

genomeClustering(RMS.obj) -> genome.clust

saveRDS(genome.clust, file = "RMSgenomeclust_all.rds")
```

Appendix E: Pre-processing Script

The basis of this script is from the Readme.md file for the *microrms*-package (Snipen, 2021)

```
#!/bin/bash

#SBATCH --array=1-64%8
#SBATCH --nodes=1
#SBATCH --ntasks=10                     # The number of threads reserved
#SBATCH --mem=30G                       # The amount of memory reserved
#SBATCH --partition=smallmem,hugemem    # For < 100GB use smallmem, for >100GB use hugemem
#SBATCH --time=10:00:00                 # Wall time
#SBATCH --job-name=rms_reads            # Sensible name for the job
#SBATCH --output=rms_reads_%j_%a.log    # Logfile output here


###############
### Settings
###
threads=10
metadata_file=<Path to sorted metadata-file>
fastq_folder=<Folder with input fastq files>
fasta_folder=fasta              # folder for output fasta files
tmp_folder=tmp                  # temporary output folder
forward_primer_length=16        # GACTGCGTACCAATTC
reverse_primer_length=16        # GATGAGTCCTGAGTAA
min_read_length=30
maxee=0.02
if [ ! -d $fasta_folder ]
then
  mkdir $fasta_folder
fi
if [ ! -d $tmp_folder ]
then
  mkdir $tmp_folder
fi


##################
### The metadata
###
SampleID=$(awk 'NR==1{for(i=1;i<=NF;i++){f[$i] = i}}{print $(f["SampleID"])}' $metadata_file)
R1_files=$(awk 'NR==1{for(i=1;i<=NF;i++){f[$i] = i}}{print $(f["filename"])}' $metadata_file)
R2_files=$(awk 'NR==1{for(i=1;i<=NF;i++){f[$i] = i}}{print $(f["filename2"])}' $metadata_file)


line=$(($SLURM_ARRAY_TASK_ID+1))
sid=$(echo $SampleID | awk -vidx=$line '{print $idx}')
r1=$fastq_folder/$(echo $R1_files | awk -vidx=$line '{print $idx}')
r2=$fastq_folder/$(echo $R2_files | awk -vidx=$line '{print $idx}')
echo "Processing sample $sid"
echo "R1 file $r1"
echo "R2 file $r2"


#############################
### The processing
###
echo VSEARCH quality filtering sample...
singularity exec /cvmfs/singularity.galaxyproject.org/v/s/vsearch:2.18.0--h95f258a_0 vsearch \
 --fastq_filter $r1 \
 --reverse $r2 \
 --fastq_maxee_rate $maxee \
 --fastqout $tmp_folder/$sid\_filtered_R1.fq \
 --fastqout_rev $tmp_folder/$sid\_filtered_R2.fq
```

```
echo VSEARCH mergings read-pairs...
singularity exec /cvmfs/singularity.galaxyproject.org/v/s/vsearch:2.18.0--h95f258a_0 vsearch \
 --threads $threads \
 --fastq_mergepairs $tmp_folder/$sid\_filtered_R1.fq \
 --reverse $tmp_folder/$sid\_filtered_R2.fq \
 --fastq_allowmergestagger --fastq_minmergelen $min_read_length \
 --fastaout $tmp_folder/$sid\_merged.fa \
 --fastqout_notmerged_fwd $tmp_folder/$sid\_notmerged_R1.fq \
 --fastqout_notmerged_rev $tmp_folder/$sid\_notmerged_R2.fq

echo VSEARCH trimming primers from merged reads...
singularity exec /cvmfs/singularity.galaxyproject.org/v/s/vsearch:2.18.0--h95f258a_0 vsearch \
 --fastx_filter $tmp_folder/$sid\_merged.fa \
 --fastq_stripleft $forward_primer_length \
 --fastq_stripright $reverse_primer_length \
 --fastq_minlen $min_read_length \
 --relabel 'size=2;pair' \
 --fastaout $tmp_folder/$sid\_merged_trim.fa

echo VSEARCH trimming primers from un-merged reads...
singularity exec /cvmfs/singularity.galaxyproject.org/v/s/vsearch:2.18.0--h95f258a_0 vsearch \
 --fastq_filter $tmp_folder/$sid\_notmerged_R1.fq \
 --fastq_stripleft $forward_primer_length \
 --fastq_minlen $min_read_length \
 --relabel 'size=1;notmerged_R1_' \
 --fastaout $tmp_folder/$sid\_notmerged_R1_trim.fa
singularity exec /cvmfs/singularity.galaxyproject.org/v/s/vsearch:2.18.0--h95f258a_0 vsearch \
 --fastq_filter $tmp_folder/$sid\_notmerged_R2.fq \
 --fastq_stripleft $reverse_primer_length \
 --fastq_minlen $min_read_length \
 --relabel 'size=1;notmerged_R2_' \
 --fastaout $tmp_folder/$sid\_notmerged_R2_trim.fa


echo VSEARCH adding all reads to one fasta-file...
cat $tmp_folder/$sid\_notmerged_R1_trim.fa >> $tmp_folder/$sid\_merged_trim.fa
singularity exec /cvmfs/singularity.galaxyproject.org/v/s/vsearch:2.18.0--h95f258a_0 vsearch \
 --fastx_revcomp $tmp_folder/$sid\_notmerged_R2_trim.fa \
 --fastaout $tmp_folder/$sid\_notmerged_R2_trim_rc.fa
cat $tmp_folder/$sid\_notmerged_R2_trim_rc.fa >> $tmp_folder/$sid\_merged_trim.fa

echo VSEARCH de-replicating...
singularity exec /cvmfs/singularity.galaxyproject.org/v/s/vsearch:2.18.0--h95f258a_0 vsearch \
 --threads $threads \
 --derep_fulllength $tmp_folder/$sid\_merged_trim.fa \
 --minuniquesize 1 \
 --minseqlength $min_read_length \
 --sizein --sizeout \
 --relabel $sid:uread_ \
 --output $fasta_folder/$sid.fasta


###############
### Cleaning
###
rm $tmp_folder/$sid\_filtered_R1.fq
rm $tmp_folder/$sid\_filtered_R2.fq
rm $tmp_folder/$sid\_notmerged_R1.fq
rm $tmp_folder/$sid\_notmerged_R2.fq
rm $tmp_folder/$sid\_notmerged_R1_trim.fa
rm $tmp_folder/$sid\_notmerged_R2_trim.fa
rm $tmp_folder/$sid\_notmerged_R2_trim_rc.fa
rm $tmp_folder/$sid\_merged.fa
rm $tmp_folder/$sid\_merged_trim.fa
```

American Society for Microbiology. (2003). *Microbial Communities: Advantages of Multicellular Cooperation* (Microbial Communities: From Life Apart to Life Together, Issue. https://www.ncbi.nlm.nih.gov/books/NBK562920/

Baker, G. C., Smith, J. J., & Cowan, D. A. (2003). Review and re-analysis of domain-specific 16S primers. *J Microbiol Methods*, *55*(3), 541-555. https://doi.org/10.1016/j.mimet.2003.08.009

Breitwieser, F. P., Lu, J., & Salzberg, S. L. (2019). A review of methods and databases for metagenomic classification and assembly. *Brief Bioinform*, *20*(4), 1125-1136. https://doi.org/10.1093/bib/bbx120

Cermak, N., Datta, M. S., & Conwill, A. (2020). Rapid, Inexpensive Measurement of Synthetic Bacterial Community Composition by Sanger Sequencing of Amplicon Mixtures. *iScience*, *23*(3), 100915. https://doi.org/https://doi.org/10.1016/j.isci.2020.100915

Cruz-Morales, P., Orellana, C. A., Moutafis, G., Moonen, G., Rincon, G., Nielsen, L. K., & Marcellin, E. (2019). Revisiting the Evolution and Taxonomy of Clostridia, a Phylogenomic Update. *Genome Biology and Evolution*, *11*(7), 2035-2044. https://doi.org/10.1093/gbe/evz096

Díaz, R., Torres-Miranda, A., Orellana, G., & Garrido, D. (2021). Comparative Genomic Analysis of Novel Bifidobacterium longum subsp. longum Strains Reveals Functional Divergence in the Human Gut Microbiota. *Microorganisms*, *9*(9), 1906. https://www.mdpi.com/2076-2607/9/9/1906

Duranti, S., Milani, C., Lugli, G. A., Mancabelli, L., Turroni, F., Ferrario, C., Mangifesta, M., Viappiani, A., Sánchez, B., Margolles, A., van Sinderen, D., & Ventura, M. (2016). Evaluation of genetic diversity among strains of the human gut commensal Bifidobacterium adolescentis. *Scientific Reports*, *6*(1), 23971. https://doi.org/10.1038/srep23971

Favier, C. F., Vaughan, E. E., De Vos, W. M., & Akkermans, A. D. (2002). Molecular monitoring of succession of bacterial communities in human neonates. *Appl Environ Microbiol*, *68*(1), 219-226. https://doi.org/10.1128/aem.68.1.219-226.2002

Federhen, S. (2011). The NCBI Taxonomy database. *Nucleic Acids Research*, *40*(D1), D136-D143. https://doi.org/10.1093/nar/gkr1178

Gill, S. R., Pop, M., DeBoy, R. T., Eckburg, P. B., Turnbaugh, P. J., Samuel, B. S., Gordon, J. I., Relman, D. A., Fraser-Liggett, C. M., & Nelson, K. E. (2006). Metagenomic Analysis of the Human Distal Gut Microbiome. *Science*, *312*(5778), 1355-1359. https://doi.org/doi:10.1126/science.1124234

Goodwin, S., McPherson, J. D., & McCombie, W. R. (2016). Coming of age: ten years of next-generation sequencing technologies. *Nature Reviews Genetics*, *17*(6), 333-351. https://doi.org/10.1038/nrg.2016.49

Handelsman, J., Rondon, M. R., Brady, S. F., Clardy, J., & Goodman, R. M. (1998). Molecular biological access to the chemistry of unknown soil microbes: a new frontier for natural products. *Chemistry & Biology*, *5*(10), R245-R249. https://doi.org/https://doi.org/10.1016/S1074-5521(98)90108-9

Hess, M. K., Rowe, S. J., Van Stijn, T. C., Henry, H. M., Hickey, S. M., Brauning, R., McCulloch, A. F., Hess, A. S., Kirk, M. R., Kumar, S., Pinares-Patiño, C., Kittelmann, S., Wood, G. R., Janssen, P. H., & McEwan, J. C. (2020). A restriction enzyme reduced representation sequencing approach for low-cost, high-throughput metagenome profiling. *PLOS ONE*, *15*(4), e0219882. https://doi.org/10.1371/journal.pone.0219882

Hiseni, P., Rudi, K., Wilson, R. C., Hegge, F. T., & Snipen, L. (2021). HumGut: a comprehensive human gut prokaryotic genomes collection filtered by metagenome data. *Microbiome*, *9*(1), 165. https://doi.org/10.1186/s40168-021-01114-w

Hugenholtz, P., Skarshewski, A., & Parks, D. H. (2016). Genome-Based Microbial Taxonomy Coming of Age. *Cold Spring Harb Perspect Biol*, *8*(6). https://doi.org/10.1101/cshperspect.a018085

Janda, J. M., & Abbott Sharon, L. (2007). 16S rRNA Gene Sequencing for Bacterial Identification in the Diagnostic Laboratory: Pluses, Perils, and Pitfalls. *Journal of Clinical Microbiology*, *45*(9), 2761-2764. https://doi.org/10.1128/JCM.01228-07

Janda, J. M., & Abbott, S. L. (2007). 16S rRNA gene sequencing for bacterial identification in the diagnostic laboratory: pluses, perils, and pitfalls. *J Clin Microbiol*, *45*(9), 2761-2764. https://doi.org/10.1128/jcm.01228-07

Killingstad, M.-E. (2021). *Colonization of Bifidobacterium in the Human Infant Gut* Norwegian University of Lice Sciences]. Ås, Norway. https://nmbu.brage.unit.no/nmbu-xmlui/handle/11250/2835681

Laursen, M. F., Sakanaka, M., von Burg, N., Mörbe, U., Andersen, D., Moll, J. M., Pekmez, C. T., Rivollier, A., Michaelsen, K. F., Mølgaard, C., Lind, M. V., Dragsted, L. O., Katayama, T., Frandsen, H. L., Vinggaard, A. M., Bahl, M. I., Brix, S., Agace, W., Licht, T. R., & Roager, H. M. (2021). Bifidobacterium species associated with breastfeeding produce aromatic lactic acids in the infant gut. *Nature Microbiology*, *6*(11), 1367-1382. https://doi.org/10.1038/s41564-021-00970-4

Liland, K. H., Vinje, H., & Snipen, L. (2017). microclass: an R-package for 16S taxonomy classification. *BMC Bioinformatics*, *18*(1), 172. https://doi.org/10.1186/s12859-017-1583-2

Liland, K. H., Vinje, Hilde. & Snipen, L. (2021). *microclass: Tools for taxonomic classification of prokaryotes*. In

Liu, M. Y., Worden, P., Monahan, L. G., DeMaere, M. Z., Burke, C. M., Djordjevic, S. P., Charles, I. G., & Darling, A. E. (2017). Evaluation of ddRADseq for reduced representation metagenome sequencing. *PeerJ*, *5*, e3837. https://doi.org/10.7717/peerj.3837

Lødrup Carlsen, K. C., Rehbinder, E. M., Skjerven, H. O., Carlsen, M. H., Fatnes, T. A., Fugelli, P., Granum, B., Haugen, G., Hedlin, G., Jonassen, C. M., Landrø, L., Lunde, J., Marsland, B. J., Nordlund, B., Rudi, K., Sjøborg, K., Söderhäll, C., Staff, A. C., Vettukattil, R., & Carlsen, K. H. (2018). Preventing Atopic Dermatitis and ALLergies in Children-the PreventADALL study. *Allergy*, *73*(10), 2063-2070. https://doi.org/10.1111/all.13468

Makino, H., Kushiro, A., Ishikawa, E., Kubota, H., Gawad, A., Sakai, T., Oishi, K., Martin, R., Ben-Amor, K., Knol, J., & Tanaka, R. (2013). Mother-to-infant transmission of intestinal bifidobacterial strains has an impact on the early development of vaginally delivered infant's microbiota. *PLOS ONE*, *8*(11), e78331. https://doi.org/10.1371/journal.pone.0078331

Massello, F. L., Chan, C. S., Chan, K.-G., Goh, K. M., Donati, E., & Urbieta, M. S. (2020). Meta-Analysis of Microbial Communities in Hot Springs: Recurrent Taxa and Complex Shaping Factors beyond pH and Temperature. *Microorganisms*, *8*(6), 906. https://www.mdpi.com/2076-2607/8/6/906

Naghoni, A., Emtiazi, G., Amoozegar, M. A., Cretoiu, M. S., Stal, L. J., Etemadifar, Z., Shahzadeh Fazeli, S. A., & Bolhuis, H. (2017). Microbial diversity in the hypersaline Lake Meyghan, Iran. *Scientific Reports*, *7*(1), 11522. https://doi.org/10.1038/s41598-017-11585-3

National Human Genome Research Institute. (2021). The Cost of Sequencing a Human Genome. https://www.genome.gov/about-genomics/fact-sheets/Sequencing-Human-Genome-cost

Nilsen, M. (2022). *The gut microbiota from the general population during the first year of life* [PhD, thesis number 2022:67, Norwegian University of Life Sciences]. Ås, Norway.

O'Leary, N. A., Wright, M. W., Brister, J. R., Ciufo, S., Haddad, D., McVeigh, R., Rajput, B., Robbertse, B., Smith-White, B., Ako-Adjei, D., Astashyn, A., Badretdin, A., Bao, Y., Blinkova, O., Brover, V., Chetvernin, V., Choi, J., Cox, E., Ermolaeva, O., . . . Pruitt, K. D. (2016). Reference sequence (RefSeq) database at NCBI: current status, taxonomic expansion, and functional annotation. *Nucleic Acids Res*, *44*(D1), D733-745. https://doi.org/10.1093/nar/gkv1189

R Development Core Team. (2021). *R: A Language and Environment for Statistical Computing*. In R Foundation for Statistical Computing. https://www.R-project.org

Ravi, A., Avershina, E., Angell, I. L., Ludvigsen, J., Manohar, P., Padmanaban, S., Nachimuthu, R., Snipen, L., & Rudi, K. (2018). Comparison of reduced metagenome and 16S rRNA gene sequencing for

determination of genetic diversity and mother-child overlap of the gut associated microbiota. *J Microbiol Methods*, *149*, 44-52. https://doi.org/10.1016/j.mimet.2018.02.016

Ravi, A., Avershina, E., Angell, I. L., Ludvigsen, J., Manohar, P., Padmanaban, S., Nachimuthu, R., Snipen, L., & Rudi, K. (2018). Comparison of reduced metagenome and 16S rRNA gene sequencing for determination of genetic diversity and mother-child overlap of the gut associated microbiota. *Journal of Microbiological Methods*, *149*, 44-52. https://doi.org/https://doi.org/10.1016/j.mimet.2018.02.016

Reinartz, J., Bruyns, E., Lin, J. Z., Burcham, T., Brenner, S., Bowen, B., Kramer, M., & Woychik, R. (2002). Massively parallel signature sequencing (MPSS) as a tool for in-depth quantitative gene expression profiling in all organisms. *Brief Funct Genomic Proteomic*, *1*(1), 95-104. https://doi.org/10.1093/bfgp/1.1.95

Rognes, T., Flouri, T., Nichols, B., Quince, C., & Mahé, F. (2016). VSEARCH: a versatile open source tool for metagenomics. *PeerJ*, *4*, e2584. https://doi.org/10.7717/peerj.2584

Schmalenberger, A., Schwieger, F., & Tebbe, C. C. (2001). Effect of primers hybridizing to different evolutionarily conserved regions of the small-subunit rRNA gene in PCR-based microbial community analyses and genetic profiling. *Appl Environ Microbiol*, *67*(8), 3557-3563. https://doi.org/10.1128/aem.67.8.3557-3563.2001

Schoch, C. L., Ciufo, S., Domrachev, M., Hotton, C. L., Kannan, S., Khovanskaya, R., Leipe, D., Mcveigh, R., O'Neill, K., Robbertse, B., Sharma, S., Soussov, V., Sullivan, J. P., Sun, L., Turner, S., & Karsch-Mizrachi, I. (2020). NCBI Taxonomy: a comprehensive update on curation, resources and tools. *Database*, *2020*. https://doi.org/10.1093/database/baaa062

Sims, D., Sudbery, I., Ilott, N. E., Heger, A., & Ponting, C. P. (2014). Sequencing depth and coverage: key considerations in genomic analyses. *Nature Reviews Genetics*, *15*(2), 121-132. https://doi.org/10.1038/nrg3642

Snipen, L. (2022). microrms: Processing and analysis of RMS data. https://github.com/larssnip/microRMS.

Snipen, L., Angell, I. L., Rognes, T., & Rudi, K. (2021). Reduced metagenome sequencing for strain-resolution taxonomic profiles. *Microbiome*, *9*(1), 79. https://doi.org/10.1186/s40168-021-01019-8

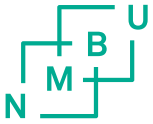
Snipen, L., & Liland, K. H. (2021). *microseq: Basic Biological Sequence Handling*. In https://CRAN.R-project.org/package=microseq

Stewart, E. J. (2012). Growing Unculturable Bacteria. *Journal of Bacteriology*, *194*(16), 4151-4160. https://doi.org/doi:10.1128/JB.00345-12

Wickham, H. (2016). *ggplot2: Elegant Graphics for Data Analysis*. Springer-Verlag New York. https://ggplot2.tidyverse.org

Wong, C. B., Odamaki, T., & Xiao, J.-z. (2020). Insights into the reason of Human-Residential Bifidobacteria (HRB) being the natural inhabitants of the human gut and their potential health-promoting benefits. *FEMS Microbiology Reviews*, *44*(3), 369-385. https://doi.org/10.1093/femsre/fuaa010

Yao, S., Zhao, Z., Wang, W., & Liu, X. (2021). Bifidobacterium Longum: Protection against Inflammatory Bowel Disease. *J Immunol Res*, *2021*, 8030297. https://doi.org/10.1155/2021/8030297



Norges miljø- og biovitenskapelige universitet
Noregs miljø- og biovitskapelege universitet
Norwegian University of Life Sciences

Postboks 5003
NO-1432 Ås
Norway